NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

STUDY OF SOFTWARE TOOLS TO SUPPORT SYSTEMS ENGINEERING MANAGEMENT

by

Peter Shchupak

June 2015

Thesis Advisor:	Charles Pickar
Second Reader:	Paul Shebalin

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2015	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE STUDY OF SOFTWARE TOOLS TO SUPPORT SYSTEMS ENGINEERING MANAGEMENT	**5. FUNDING NUMBERS**
6. AUTHOR(S) Peter Shchupak	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School, Monterey, CA 93943–5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Space and Naval Warfare Systems Center Pacific, 53560 Hull St, San Diego, CA 92152	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

According to a 2010 Government Accountability Office (GAO) report, major system acquisitions within the Department of Defense (DOD) tend to be behind schedule, over budget, and often fail to deliver at least some of the planned capabilities. One area that can significantly contribute to successful implementation of systems engineering is the regular usage of management software tools and their continued evolution to better meet systems engineering needs. This thesis provides a detailed exploration of four categories of available system engineering management tools: Model-Based Systems Engineering (MBSE), Product Life Cycle Management (PLM), Systems Engineering Environment (SEE), and Project Management software. Each tool has numerous features that support successful systems engineering. However, there does not seem to be a consolidated commercially available tool or system that allows for seamless management of systems engineering projects across all of the process areas. Drawing upon these existing tools and the International Council on Systems Engineering (INCOSE) processes, this thesis derives a set of requirements for such a consolidated systems engineering management tool. This research can serve as the starting point for a follow-on effort to develop such a tool.

14. SUBJECT TERMS Systems engineering management, systems engineering processes, Model-Based Systems Engineering (MBSE), Product Lifecycle Management (PLM), Systems Engineering Environment (SEE), Project Management	**15. NUMBER OF PAGES** 137
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540–01–280–5500 Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

STUDY OF SOFTWARE TOOLS TO SUPPORT SYSTEMS ENGINEERING MANAGEMENT

Peter Shchupak
Space and Naval Warfare Systems Center Pacific
M.Eng., University of California San Diego, 2005

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
June 2015

Author: Peter Shchupak

Approved by: Charles Pickar, Ph.D.
Thesis Advisor

Paul Shebalin, Sc.D.
Second Reader

Cliff Whitcomb, Ph.D.
Chair, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

According to a 2010 Government Accountability Office (GAO) report, major system acquisitions within the Department of Defense (DOD) tend to be behind schedule, over budget, and often fail to deliver at least some of the planned capabilities. One area that can significantly contribute to successful implementation of systems engineering is the regular usage of management software tools and their continued evolution to better meet systems engineering needs. This thesis provides a detailed exploration of four categories of available system engineering management tools: Model-Based Systems Engineering (MBSE), Product Life Cycle Management (PLM), Systems Engineering Environment (SEE), and Project Management software. Each tool has numerous features that support successful systems engineering. However, there does not seem to be a consolidated commercially available tool or system that allows for seamless management of systems engineering projects across all of the process areas. Drawing upon these existing tools and the International Council on Systems Engineering (INCOSE) processes, this thesis derives a set of requirements for such a consolidated systems engineering management tool. This research can serve as the starting point for a follow-on effort to develop such a tool.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AMSMP	Acquisition M&S Master Plan
BOM	Bill of Materials
CAD	Computer-aided design
CM	Configuration Management
COTS	Commercial-Off-the-Shelf
CSE	Chief Systems Engineer
DAG	Defense Acquisition Guide
DAU	Defense Acquisition University
DCMA	Defense Contract Management Agency
DISA	Defense Information Systems Agency
DOD	Department of Defense
DODAF	Department of Defense Architecture Framework
ECR	Engineering Change Request
ePLM IDE	Enterprise Product Life cycle Management Integrated Data Environment
ERP	Enterprise Resource Planning
EVM	Earned Value Management
FFBD	Functional Flow Block Diagram
GOTS	Government-Off-the-Shelf
HSI	Human Systems Integration
IBM	International Business Machines
ICD	Interface Control Document
IDE	Integrated Data Environment
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
ILS	Integrated Logistics Support
INCOSE	International Council on Systems Engineering
ISEE	Integrated Systems Engineering Environment
ISO	International Organization for Standardization
IT	Information Technology

MBE	Model-Based Engineering
MBSE	Model-Based Systems Engineering
MODAF	Ministry of Defence Architecture Framework
NAVSEA	Naval Sea Systems Command
NDIA	National Defense Industrial Association
NTW TBMD	Navy Theater Wide Theater Ballistic Missile Defense
OSEE	Open Systems Engineering Environment
PDM	Product Data Management
PEO IWS	Program Executive Office Integrated Warfare Systems
PLM	Product Life cycle Management
POA&M	Plan of Action and Milestones
QA	Quality Assurance
RVTM	Requirements Verification Traceability Matrix
SDEA	System Definition-Enabled Acquisition
SECD	Systems Engineering Concept Demonstration
SEE	Systems Engineering Environment
SEP	Systems Engineering Plan
SLIM	Systems Lifecycle Management
SoS	System of Systems
SOW	Statement of Work
TSM	Total System Model
VV&A	Verification, Validation, and Accreditation

EXECUTIVE SUMMARY

According to a 2010 GAO report, major system acquisitions within the Department of Defense (DOD) tend to be behind schedule, over budget, and often fail to deliver at least some of the planned capabilities (GAO 2010, under "Highlights"). With decreasing DOD budgets and increased oversight there is growing pressure to address these issues. In their 2008 Report on Systemic Root Cause Analysis of Program Failures the National Defense Industrial Association (NDIA) "recognize(d) that there is a strong relationship between disciplined systems engineering and good management decision making in the critical early states of an acquisition cycle" (NDIA 2008, 3). One area that can significantly contribute to successful implementation of systems engineering is the regular usage of systems engineering management software tools and as updated to better meet systems engineering needs. This thesis explores the key components of systems engineering management, conducts a survey of existing software tools that can be used to support systems engineering management, and proposes requirements for a tool that would improve systems engineering management.

This thesis finds that although there are a variety of software products available to support systems engineering management, they do not seamlessly integrate to support a systems engineering effort from beginning to end. This thesis recommends that developing a single consolidated tool or a suite of integrated tools to support the systems engineering management effort would significantly benefit the systems engineering community. And, in turn, it would significantly benefit the DOD in executing highly complex systems engineering efforts. However, it seems that the DOD has not yet started adopting Systems Engineering Environment (SEE) types of tool sets. It would be advantageous for the DOD to put a focus on moving in this direction. This in turn could motivate industry to spend more resources in producing a product that could act as the glue for guiding a systems engineering effort. The starting point for developing such a product is recommended to be the set of International Council on Systems Engineering (INCOSE) or Defense Acquisition University (DAU) processes.

This thesis provides a survey of four different categories of software tools that could support systems engineering management. Each category is described and the benefits and challenges are discussed. The first category is Model-Based Systems Engineering (MBSE). It is a highly process-focused technique that parallels the systems engineering processes. INCOSE predicts that MBSE will be fully mature and ready for full adoption at the organizational level by 2020, and there are DOD efforts underway to embrace MBSE. The second category is Product Life Cycle Management (PLM). It is a holistic approach for managing systems engineering efforts through the entire life cycle. The DOD is looking at PLM as a solution to help deal with significant complexity and to reduce costs.

The third category is SEE. It is an integrated environment for executing systems engineering efforts throughout the life cycle. SEE seems to be a very promising concept for addressing the challenges of managing a systems engineering effort but unfortunately does not seem to have been able to gain a meaningful foothold within DOD. The final category is Project Management tools. It focuses on a range of tools that although do not directly relate to systems engineering, do have a number of features that would prove useful to any team and manager.

All four categories of tools offer features of significant benefit to a Chief Systems Engineer (CSE). Some of these tools can also be used in combination to extend those benefits (such as MBSE and PLM). And the SEE concept presents a promising approach to having a central system through which the CSE can manage the systems engineering effort. However, there currently does not seem to be a consolidated commercially available tool or system that allows for seamless management of systems engineering projects across all of the process areas.

Finally, a set of key features is listed and requirements are developed for a central tool that supports systems engineering management. The approach used is to start with the INCOSE systems engineering processes as the central guide for building such a tool. This approach supports a broad range of systems engineering efforts by allowing for significant tailoring. The requirements are derived from the activities and sub-activities described for each process. Several key stipulations are offered. First, the management

tool is intended to be a guide for the CSE and not a replacement for activities and decisions that must still be made by humans. Second, the set of requirements is not an exhaustive set but is intended as a starting point.

The envisioned systems engineering management tool would leverage the benefits of existing tools by either integrating with them or offering similar functionality. There are three areas where the tool would be especially beneficial. The first would be to provide a standardized approach to managing a systems engineering effort by guiding it from start to finish. This would help normalize for experience level of the CSE and would also reduce dependence on one or a few key individuals. The second benefit is added insight into progress and challenges for the CSE, management, and decision makers by captured real-time status of the project. The third benefit is more complete and reliable organizational knowledge transfer.

There is significant room to further expand beyond the set of requirements developed in this thesis, and one improvement could be to obtain feedback from practicing CSEs. The next step would be to create a prototype systems engineering management tool that can be tested on a real project.

References

Government Accountability Office (GAO). 2010. *Managing Risk to Achieve Better Outcomes*. GAO-10–374T. Washington, DC: U.S. Government Accountability Office. http://www.gao.gov/assets/130/123946.pdf.

National Defense Industrial Association (NDIA). 2008. *Report on Systemic Root Cause Analysis Of Program Failures*. Washington, DC: U.S. Department of Defense. http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Studies/NDIASRCAReportFINA18Dec2008.pdf.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank my advisor, Charles Pickar, for the feedback, support, and patience as I worked through my research and writing.

I thank my wife, Joy, for standing by me through this entire process and giving me the support and motivation to carry on. I would also like to thank my daughter, Izzy, for taking her naps and allowing me the time to write. And I would like to thank my parents, Leo and Irina, for instilling in me the drive to work hard and overcome challenges.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

According to a 2010 GAO report, major system acquisitions within the Department of Defense (DOD) tend to be behind schedule, over budget, and often fail to deliver at least some of the planned capabilities (GAO 2010, under “Highlights”). With decreasing DOD budgets there is growing pressure to address these issues. In their 2008 Report on Systemic Root Cause Analysis of Program Failures, the National Defense Industrial Association (NDIA) “recognize(d) that there is a strong relationship between disciplined systems engineering and good management decision making in the critical early states of an acquisition cycle” (NDIA 2008, 3). One area that can significantly contribute to successful implementation of systems engineering is the regular usage of systems engineering management software tools and their continued evolution to better meet systems engineering needs. This thesis will explore the key components of systems engineering management, conduct a survey of existing software tools that can be used to support systems engineering management, and propose requirements for a tool that would facilitate systems engineering management.

B. RESEARCH QUESTIONS

This thesis explores the three following questions.

1. What are the key components of systems engineering management?

The first step of this study is to explore the key components of systems engineering management. Systems engineering teaches that before a solution can be developed the underlying problem must be fully understood. The solution must then trace from this deeper understanding, thereby validating that the solution is indeed the correct one for the problem at hand. Therefore, when searching for a way to improve the management of systems engineering efforts, it is critical to first explore what systems engineering management entails.

2. What software tools are available that could support systems engineering management?

It is prudent to perform a survey of available tools that could support systems engineering management. The goal is to leverage and build upon existing solutions. Furthermore, an appropriate solution may already exist thereby leading to an endorsement of a particular tool category. Since there are numerous individual tools, the approach taken will be to explore tool categories and identify the general benefits and challenges for each category.

3. What requirements would an ideal systems engineering management tool have?

This final question explores the key features for a software tool to support systems engineering management. It builds upon the results of question one and is further informed by the results of question two.

C. SYSTEMS ENGINEERING CHALLENGES

In the early 1990s, the Air Force funded the Systems Engineering Concept Demonstration (SECD) to "demonstrate the concept of an advanced computer-based environment of integrated software tools and methods which supports the…systems life cycle" with the intent that "systems and specialty engineers can increase their productivity and effectiveness during the development, maintenance, and enhancement of military computer-based systems" (Comer and Rohde 1992, 3). This was "one of the first efforts to seriously address automation of the systems engineering process" (Comer and Rohde 1992, 4), motivated by the realization of both the importance and difficulty of the systems engineering role in complex projects. The study organized systems engineering activities into three categories: engineering, communication, and management. It then listed needs and problems in each category. The underlying theme supported the thesis that in each area there was a significant need for automated support. In the management category specifically, the need for automated support was identified for the areas of process management, program planning and management, and task management. The communication category lists automation needs in the areas of collaboration and coordination, boundary spanning, and joint work product development.

Computer technology has experienced tremendous growth since the SECD study and many systems engineering automation tools are now available. However, in a 2010 report on the top systems engineering issues NDIA highlights lack of consistent use of the latest practices and tools in the systems engineering community as well as the need for continued improvement and optimization of these software tools (Table 1). This leaves the systems engineering community exposed to many of the same challenges as they faced during the time of the SECD study.

Table 1. Top 2006 and 2010 Systems Engineering Issues (after NDIA 2010, 2).

2006 Issue	2010 Issue
Key systems engineering practices known to be effective are not consistently applied across all phases of the program life cycle.	Institutionalization of practices has shown value when adopted, but adoption tends to be spotty.
Collaborative environments, including systems engineering tools, are inadequate to effectively execute systems engineering at the joint capability, systems of systems (SoS), and system levels.	State of the practice techniques not widely utilized. Multiple tools are available but little guidance on preference exists.

The report also highlights as one of the top five systems engineering issues of 2010: "It is difficult to use currently available standard systems engineering tools early in the life cycle. In addition, many tools are not readily available and the engineers have not been trained in their use" (NDIA 2010, 6).

These issues combine to tell the story of a practice that is quickly evolving but has not yet fully matured. Ideally, systems engineers would consistently leverage standardized processes that are supported by comprehensive and integrated support tools in order to repeatedly produce high-quality products. Getting to this point is as much a systems engineering management challenge as it is a technical one. The good news is that in many respects it is possible to address both the management and technical perspectives with the same tool, or integrated suite of tools. Although the focus of this study is to identify systems engineering management tool solutions, systems engineering is also a

technical discipline so the lines between management and technical are significantly blurred. This assertion is supported by the following from the Handbook of Systems Engineering and Management: “Systems engineering involves a technical part and a managerial part. That is, it requires making technical decisions and trade-offs while controlling and managing the efforts of different experts and teams from various disciplines” (Shenhar and Sauser 2009, 120). Therefore, the ideal systems engineering management tool solution would encompass both the management and technical aspects of systems engineering.

D. BENEFITS TO SYSTEM ENGINEERING COMMUNITY

This research provides several benefits to the systems engineering community. First, this study identifies and analyzes key components of system engineering management and thereby provides an additional reference for future work in this area. Second, this study researches and reviews various categories of software management tools that can be used for systems engineering management and provides the benefits and challenges of each category. This serves to provide an organized survey of the various options that can be leveraged independently or in concert with each other to support systems engineering management. Third, it builds upon the first two items to recommend requirements of a systems engineering management tool. This analysis can be used as a starting point to develop such a tool.

E. SCOPE

This thesis surveys existing systems engineering management software tools. It reviews the key components of systems engineering management and explores systems engineering processes. It researches what management products exist that could support systems engineering management and identify the benefits and challenges of these products. Finally, it develops a set of requirements for a systems engineering management software tool. This thesis concludes with a set of tool requirements.

F. METHODOLOGY

Information on the key components of systems engineering management will be collected through literary research, online research, and personal experience.

A list of currently available software categories that can be leveraged to support systems engineering management will be gathered through literary research, online research, and personal experience. Description of each product category, as well as the benefits and challenges, will be obtained through literary and online research as well as review of existing products in that category, when appropriate.

A recommended list of systems engineering management tool requirements will be developed by the author, supported by information derived from the first two elements above as well as literary research, online research, and personal experience.

G. STRUCTURE

Chapter II Key Components for Systems Engineering Management: This chapter reviews the definition of systems engineering and highlight key management components. It then explores systems engineering processes. Finally, it looks at the typical systems engineering toolbox to identify the common tools that a systems engineer utilizes on a regular basis.

Chapter III Survey of Management Tools: This chapter reviews the various categories of management software tools and identifies the benefits and challenges associated with each. It also discusses ongoing DOD initiatives related to these categories, as applicable.

Chapter IV DOD Systems Engineering Management Tool Descriptions: This chapter describes the requirements development process for a systems engineering management software tool. It also highlights key features and benefits of such a tool.

Chapter V Conclusion and Future Research: This chapter summarizes the research and results presented in the thesis. It also presents areas that have not been fully explored in this thesis that would benefit from additional research.

Appendix: The appendix lists the systems engineering management tool requirements, and show how each requirement traces from the INCOSE systems engineering processes.

II. SYSTEMS ENGINEERING MANAGEMENT

This chapter explores the first research: What are the key components of systems engineering management? This helps lay the foundation for the remainder of the study. It does so by reviewing established systems engineering processes that form the cornerstone of systems engineering. Then it concludes with an exploration of the common software products used by CSEs for producing, gathering, and controlling information.

A. SYSTEMS ENGINEERING

Before exploring the systems engineering management process, it is necessary to review the definition of systems engineering. The International Council on Systems Engineering (INCOSE), an authoritative body on systems engineering, defines systems engineering as follows:

> Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem: Operations, Cost & Schedule, Performance, Training & Support, Test, Manufacturing, and Disposal. Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. (INCOSE 2004)

Here, one sees the focus on interdisciplinary and teaming aspects. Systems engineering requires expertise from multiple domains brought together in just the right way to develop the appropriate solution to a problem. It naturally follows that good communication is a key element for success. The definition also points out that systems engineering requires a broad perspective of the problem versus focusing on the pieces independently. This is a key consideration when looking at solutions for comprehensive management. Finally, the definition emphasizes a “structured development process” as the glue for success. The next section will explore the specifics of this process—or rather the set of processes that allow the CSE to realize this end goal.

B. SYSTEMS ENGINEERING PROCESSES

Processes contribute to a well-developed project structure and "High structure reduces the risk regardless of technology complexity or team size" (Kendrick 2009, 58). Although following a process is good practice in most undertakings regardless of complexity, it is especially important in helping navigate the complexities encountered in systems engineering efforts. A good process provides the following advantages, as noted by Tom Kendrick in "Identifying and Managing Project Risk" (Kendrick 2009, 23):

- better communications
- less rework
- lowered costs, reduced time
- earlier identification of gaps and inadequate specifications
- fewer surprises
- less chaos and firefighting.

These are all key considerations in the systems engineering realm. Another important aspect of a process is that it is repeatable and can therefore easily be applied to multiple efforts. This is the motivation for developing detailed processes and communicating them to the community of practice. This section will review established systems engineering processes by looking at two reputable sources, the INCOSE System Engineering Handbook and the Defense Acquisition Guidebook.

1. INCOSE Processes

INCOSE follows International Organization for Standardization (ISO)/ International Electrotechnical Commission (IEC) 15288:2008 and divides processes into two categories, technical and project (INCOSE 2011). The "Technical Processes…include stakeholder requirements definition, requirements analysis, architectural design, implementation, integration, verification, transition, validation, operation, maintenance, and disposal" (INCOSE 2011, 2). Technical Process definitions can be found in section 6.4 of (ISO/IEC 2008).

According to (ISO/IEC 2008, 35) these technical processes "define the activities that enable organization and project functions to optimize the benefits and reduce the

risks that arise from technical decisions and actions." In other words, they encompass the most critical technical components of systems engineering, making them the natural starting point when characterizing the key pieces of information a CSE needs access to in order to plan, manage, monitor, and make decisions.

In addition to technical processes, INCOSE also follows the ISO/IEC 15288:2008 project processes. The "Project Processes…include project planning, project assessment and control, decision management, risk management, configuration management, information management, and measurement" (INCOSE 2011, 2). Project Process definitions can be found in section 6.3 of (ISO/IEC 2008).

These processes are critical to the overall success of the project. Unlike the technical processes, the CSE does not lead the project processes, but instead contributes to them (Zipes 2007, 32). Nevertheless, the CSE must carefully track each of these as they pertain to systems engineering to ensure that appropriate insight is provided to the management team. Therefore, these processes are also an important component of the CSE's situational awareness.

Another key difference is that unlike the technical processes that occur sequentially in the more common life cycle development models, project processes "may be invoked at any time in the life cycle" (ISO/IEC 2008). This necessitates a full understanding of all of the project processes from the beginning and requires mechanisms to capture appropriate information so that it can be tracked and provided when requested.

2. Defense Acquisition University (DAU) Processes

DAU follows a similar approach to INCOSE. Processes are divided into two areas, technical processes and technical management processes (DAU 2013). The DAU technical processes, along with the purpose for each as described by DAU, are listed in Table 2.

Table 2. DAU Technical Processes (after DAU 2013, section 4.3)

Technical Processes	Purpose
Stakeholder Requirements Definition (DAU 2013, section 4.3.10)	"…helps ensure each individual stakeholder's requirements, expectations, and perceived constraints are understood from the acquisition perspective."(DAU 2013, section 4.3.10)
Requirements Analysis (DAU 2013, section 4.3.11)	"…involves the decomposition of user needs…into clear, achievable, and verifiable high-level requirements." (DAU 2013, section 4.3.11)
Architecture Design (DAU 2013, section 4.3.12)	"…allows the Program Manager and Systems Engineer to translate the outputs of the Stakeholder Requirements Definition and Requirements Analysis processes into alternative design solutions and establishes the architectural design of candidate solutions that may be found in a system model." (DAU 2013, section 4.3.12)
Implementation (DAU 2013, section 4.3.13)	"…provides a system that satisfies specified design and stakeholder performance requirements." (DAU 2013, section 4.3.13)
Integration (DAU 2013, section 4.3.14)	"…systematically assemble lower-level system elements into successively higher-level system elements, iterative with verification until the system itself emerges." (DAU 2013, section 4.3.13)
Verification (DAU 2013, section 4.3.15)	"…provides evidence that the system or system element performs its intended functions and meets all performance requirements listed in the system performance specification and functional and allocated baselines." (DAU 2013, section 4.3.15)
Validation (DAU 2013, section 4.3.16)	"…provides objective evidence that the capability provided by the system complies with stakeholder performance requirements, achieving its use in its intended operational environment." (DAU 2013, section 4.3.16)
Transition (DAU 2013, section 4.3.17)	"…process applied to move any system element to the next level in the physical architecture. For the end-item system, it is the process to install and field the system to the user in the operational environment." (DAU 2013, section 4.3.17)

The list is very similar to the INCOSE technical processes. The only difference is that DAU omits “Operation,” “Maintenance,” and “Disposal.” Instead, it seems that DAU bins each of these within the “Transition” process. A mapping between the INCOSE and DAU technical processes is provided in Figure 1.

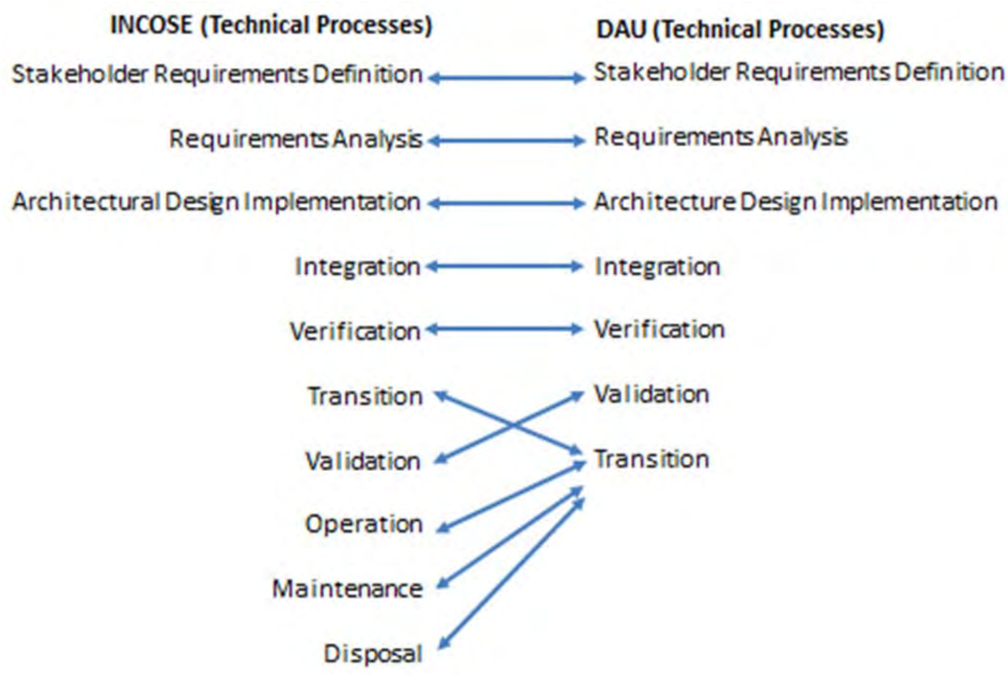

Figure 1. Mapping between INCOSE and DAU Technical Processes

Next examined are the DAU Technical Management Processes, along with the purpose for each as described by DAU (Table 3).

Table 3. DAU Technical Management Processes (after DAU 2013, section 4.3)

Technical Management Processes	**Purpose**
Technical Planning (DAU 2013, section 4.3.2)	"…provides the Program Manager and Systems Engineer with a framework to accomplish the technical activities that collectively increase product maturity and knowledge and reduce technical risks." (DAU 2013, section 4.3.2)
Decision Analysis (DAU 2013, section 4.3.3)	"…transforms a broadly stated decision opportunity into a traceable, defendable, and actionable plan." (DAU 2013, section 4.3.3)
Technical Assessment (DAU 2013, section 4.3.4)	"…allows the Systems Engineer to compare achieved results against defined criteria to provide a fact-based understanding of the current level of product knowledge, technical maturity, program status, and technical risk." (DAU 2013, section 4.3.4)
Requirements Management (DAU 2013, section 4.3.5)	"…helps ensure delivery of capability that meets intended mission performance to the operational end user." (DAU 2013, section 4.3.5)
Risk Management (DAU 2013, section 4.3.6)	"…primary method of mitigating program uncertainties and is therefore critical to achieving cost, schedule, and performance goals at every stage of the life cycle." (DAU 2013, section 4.3.6)
Configuration Management (DAU 2013, section 4.3.7)	"…allows technical insight into all levels of the system design and is the principal methodology for establishing and maintaining consistency of a system's functional, performance, and physical attributes with its requirements, design, and operational information throughout the system's life cycle." (DAU 2013, section 4.3.7)
Technical Data Management (DAU 2013, section 4.3.8)	"…identifies, acquires, manages, maintains, and ensures access to the technical data and computer software required to manage and support a system throughout the acquisition life cycle." (DAU 2013, section 4.3.8)
Interface Management (DAU 2013, section 4.3.9)	"…ensure interface definition and compliance among the system elements, as well as with other systems." (DAU 2013, section 4.3.9)

Here, one can see a slight divergence from the INCOSE approach. These processes are presented from the perspective of a systems engineer and "provide a consistent framework for managing technical activities and identifying the technical information and events critical to the success of the program" (DAU 2013). Conversely,

INCOSE takes a management perspective when presenting Project Processes, relying on input versus leadership from systems engineering. Despite this, a first order mapping between the two sets of processes can still be proposed. Although the perspectives may be different the end goal of creating a systematic approach to manage the engineering effort and support the project as a whole is the same. A mapping between the INCOSE and DAU management processes is provided in Figure 2. This mapping is developed by the author but partially informed by Lori Zipes' (2007, 23–26) presentation "Program Management vs. Systems Engineering: How different are they?" at the 10th Annual Systems Engineering Conference:

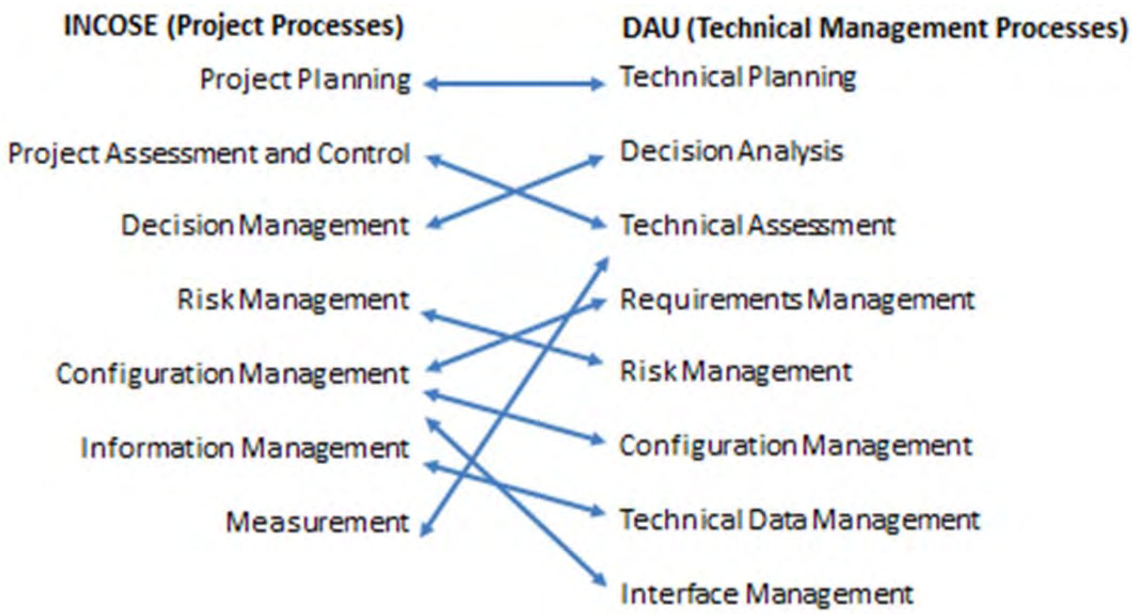

Figure 2. Mapping between INCOSE and DAU Management Processes

Lori Zipes (2007, 22) provides a good visualization of the close relationship between DAU and INCOSE processes, as well as Project Management Body of Knowledge processes (Figure 3). The diagram, along with rest of the presentation, discusses the significant overlap between systems engineering and project management functions.

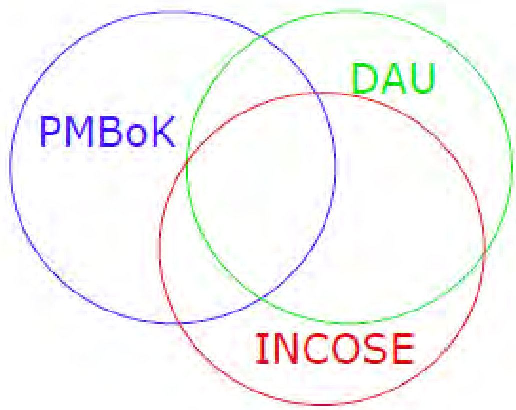

Figure 3. Process Overlap (from Zipes 2007, 22)

C. SYSTEMS ENGINEERING TOOLBOX

There are various software products that in one way or another support the systems engineering effort. Some are optimized to facilitate execution of one or more of the systems engineering processes, and others more generally support execution of a project and prove useful in managing a systems engineering effort. Table 4 is a representative list of tools that a CSE may utilize to some degree.

The pros and cons of having a large selection of tools is well described:

> The good news is that many tools are available to assist the engineer to develop solution across a wide variety of system needs. The bad news is that there is a very large selection of tools, they are not well integrated, and they are often highly tailored for narrow applications. The result is a seemingly endless landscape of un-integrated tools, methods, views, and techniques for system development. (Montgomery, Carlson, and Quartuccio 2012, 12).

The integration of information is where the real challenge rests. A presentation from an INCOSE Model-Based Systems Engineering (MBSE) workshop also highlights this challenge. It notes that the variety of tools is there but the need is for a set of tools that seamlessly covers the systems engineering Vee (Figure 4). The goal is to have a single product or a set of products that can seamlessly support a systems engineering effort from beginning to end.

Table 4. CSE Toolbox

Function	SW Tool Examples
E-mail	Microsoft Outlook, Gmail
Spreadsheet	Microsoft Excel
Presentation	Microsoft PowerPoint
Document	Microsoft Word, Adobe Acrobat
Diagram/Flowchart	Microsoft Visio
Computer-aided design (CAD)	Solidworks, Autodesk AutoCAD
Schedule	Microsoft Project, Oracle Primavera
Schedule Assessment	Booz Allen Hamilton Polaris, forProject
Earned Value Management (EVM)	Deltek Open Plan/Cobra/wInsight, Primavera P6/Cost Manager
Simulation	Mathworks MATLAB, Wolfram Mathematica
Requirements	IBM RequisitePro, IBM DOORS, Vitech CORE
Information Management	Microsoft SharePoint, TopVue
Risk Management	SwordActiveRisk Active Risk Manager, PRC Risk Register
Model-Based Systems Engineering (MBSE)	Atego Artisan Studio, 3SL Cradle, Vitech CORE
Product Life Cycle Management (PLM)	Siemens Teamcenter, PTC Windchill
Social Workflow	Sparqlight, Asana
Remote Collaboration	Defense Connect Online
Enterprise Resource Planning (ERP)	SAP ERP, Oracle ERP

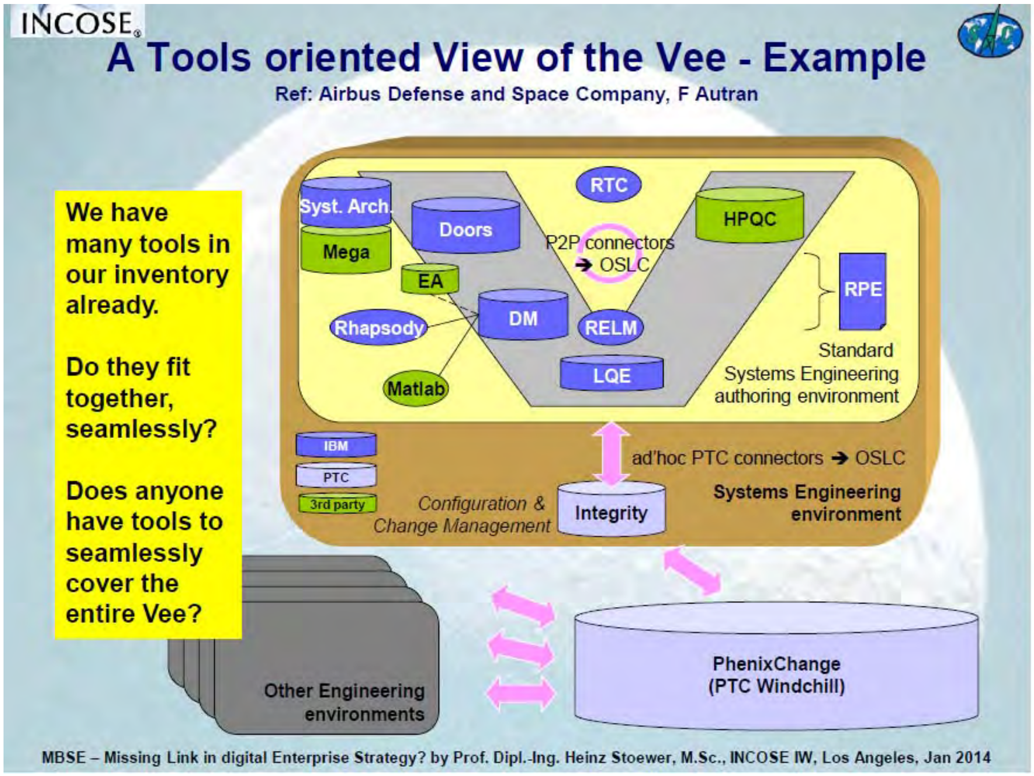

Figure 4. Tools Oriented View of the System Engineering Vee (from Heinz 2014)

D. SUMMARY

In this chapter, the key components of systems engineering management are explored. This is done by first reviewing established systems engineering processes from the perspectives of INCOSE and DAU. It is shown that both are organized by technical and management processes, and are similar. Then common software products used by CSEs for producing, gathering, and controlling information are identified. It is shown that although there are a variety of products available, they do not seamlessly integrate to support a systems engineering effort from beginning to end.

III. SURVEY OF MANAGEMENT TOOLS

This chapter provides a survey of the different types of software tools that could support systems engineering management. It looks into categories of tool and identifies the key features. It then lists the benefits and challenges. The categories that are be explored include MBSE, PLM, Systems Engineering Environment (SEE), and Project Management. Additional attention is provided to MBSE and PLM as there are ongoing initiatives within the DOD that are pushing both to the forefront.

A. MODEL-BASED SYSTEMS ENGINEERING

MBSE is defined as a "formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout the development and later life cycle phases" (Friedenthal, Greigo, and Sampson 2007, 5). The highly process-focused nature of this technique parallels the systems engineering processes discussed in Chapter II. MBSE does this by providing clear traceability between the products associated with each process. MBSE "enhances specification and design quality, reuse of system specification and design artifacts, and communications among the development team" (Friedenthal, Moore, and Steiner 2012, 15). This focus on higher quality, reduction of rework, and improved communications, as well as the process driven approach, makes MBSE a powerful tool to support systems engineering management. Several MBSE products include Atego Artisan Studio, No Magic MagicDraw, and 3SL Cradle.

The benefits of MBSE are numerous. INCOSE compiled the following list of benefits for a MBSE focused workshop (Friedenthal, Greigo, and Sampson 2007, 7):

- improved communications
- increased ability to manage system complexity
- improved product quality
- enhanced knowledge capture
- improved ability to teach and learn systems engineering fundamentals.

Management is explicitly identified as a benefit. Communications is also identified and is a key element of successful management. Improved product quality is the primary goal of good management. The others are very desirable features at the organizational level, as well as for the community of practice.

An alternate list of benefits is provided by Vitech Corporation, one of the leading MBSE product developers (Vitech Corp 2011, 112–115):

- enhanced communication
- reduced development risk
- improved quality
- increased productivity
- increased scope
- provides a structure to capture and communicate all aspects of the system
- based upon the language of the systems engineer
- contains and enforces the integrity of the system model
- latest engineering is available to the entire project team.

Communication and quality appear again on this list. Risk and scope are identified as well, both key elements that must be carefully managed for success. Increased productivity hints at a system that allows clear definition of work products and accountability for ensuring that work is done effectively and on schedule. MBSE is also designed with the systems engineering environment in mind and therefore has the benefit that it does not need to be tailored from another industry. The remaining benefits reinforce the organization and communication of information to provide a holistic view of the project in real time.

In a report on the state of Model-Based Engineering (MBE), NDIA has shown how MBE benefits map to the DOD Acquisition Life Cycle (Figure 5). It is clear that there are very significant benefits at each phase that would directly or indirectly effect cost, schedule, and performance. The report also notes that the advantages gained in the early phases also have meaningful carry over to later phases.

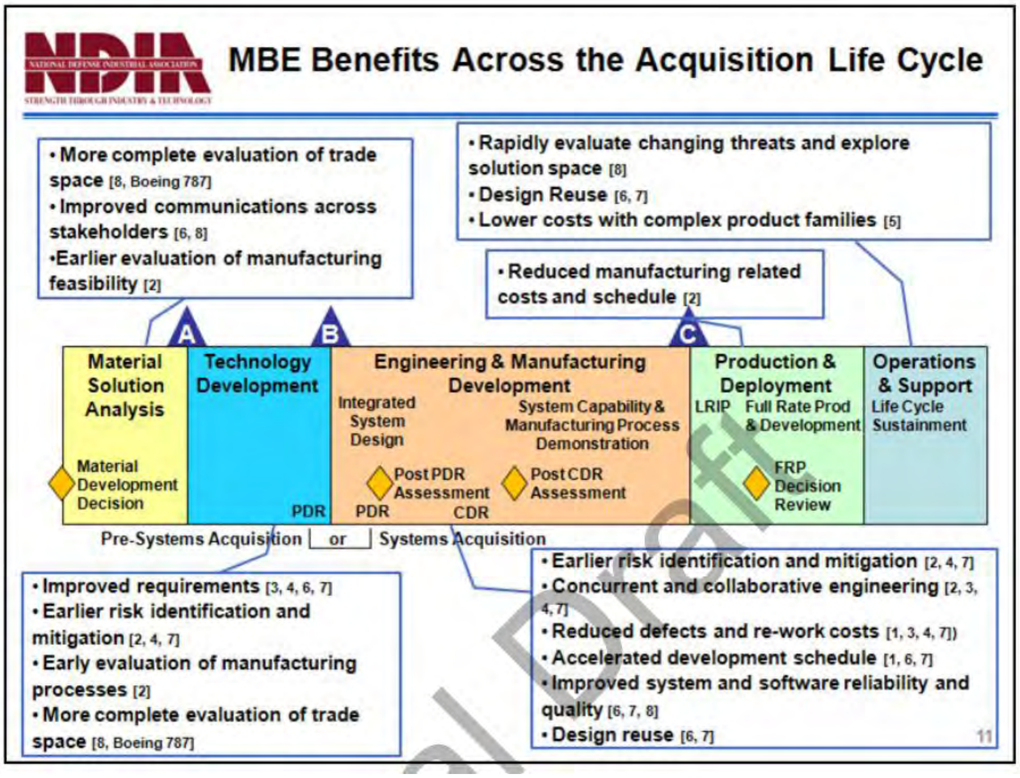

Figure 5. MBE Benefits across the Acquisition Life Cycle (from NDIA 2011, 16)

MBSE also has some challenges. A white paper developed to promote the concept of System Definition-Enabled Acquisition (SDEA) faults the current state of MBSE tools as "individually inadequate to solve the total engineering problem" (Montgomery, Carlson, and Quartuccio 2012, 17). The perspective presented is that MBSE has not yet reached an appropriate level of maturity to be the one-stop solution to systems engineering development and management. This is echoed in various other publications and forums, including at the MBSE INCOSE workshop, where two specific challenges are identified.

The first challenge is that the current state of MBSE lacks good "integration/interaction with the more 'soft' (human economics and social/environment based) elements of systems" (Heinz 2014, 28). The presentation goes on to explain that MBSE must "deal with science and art components of complex systems by also providing decision analysis support to PMs and other policy/decision makers" (Heinz

2014, 22). This hints at the need for full integration between engineering and management. In order to become a complete systems engineering solution, MBSE must incorporate the management elements along with the technical to ensure the CSE can fully execute project planning and control, and track data that must be fed up the chain to support the Project Management team. The second challenge is that “MBSE must strive to become seamless plug & play in terms of vertical and horizontal navigation between different system levels and system constituents” (Heinz 2014, 28). Currently, MBSE is just another part of the systems engineering toolbox and Heinz (2014) notes that this requires additional integration.

There are ongoing efforts to address these challenges. For example, an evolving product called Systems Lifecycle Management (SLIM) created by InterCAX attempts to fill the “gaps in current state-of-the-art commercial tools for design and analysis of complex systems” (Bajaj et al. 2011, 2) by working with what InterCAX calls the Total System Model (TSM). InterCAX describes SLIM as a “collaborative, model-based systems engineering workspace for realizing next-generation complex systems” (Bajaj et al. 2011, 1). SLIM acts as a plug-in to existing MBSE products and adds the functionality to integrate with common systems engineering software products. This integration is not only for technical tools, but also includes management tools. Figure 6 shows this integration to other functional areas and software products. The connectivity with PLM is also significant. PLM is gaining a lot of momentum as a management technique for complex projects and will be discussed in the next section.

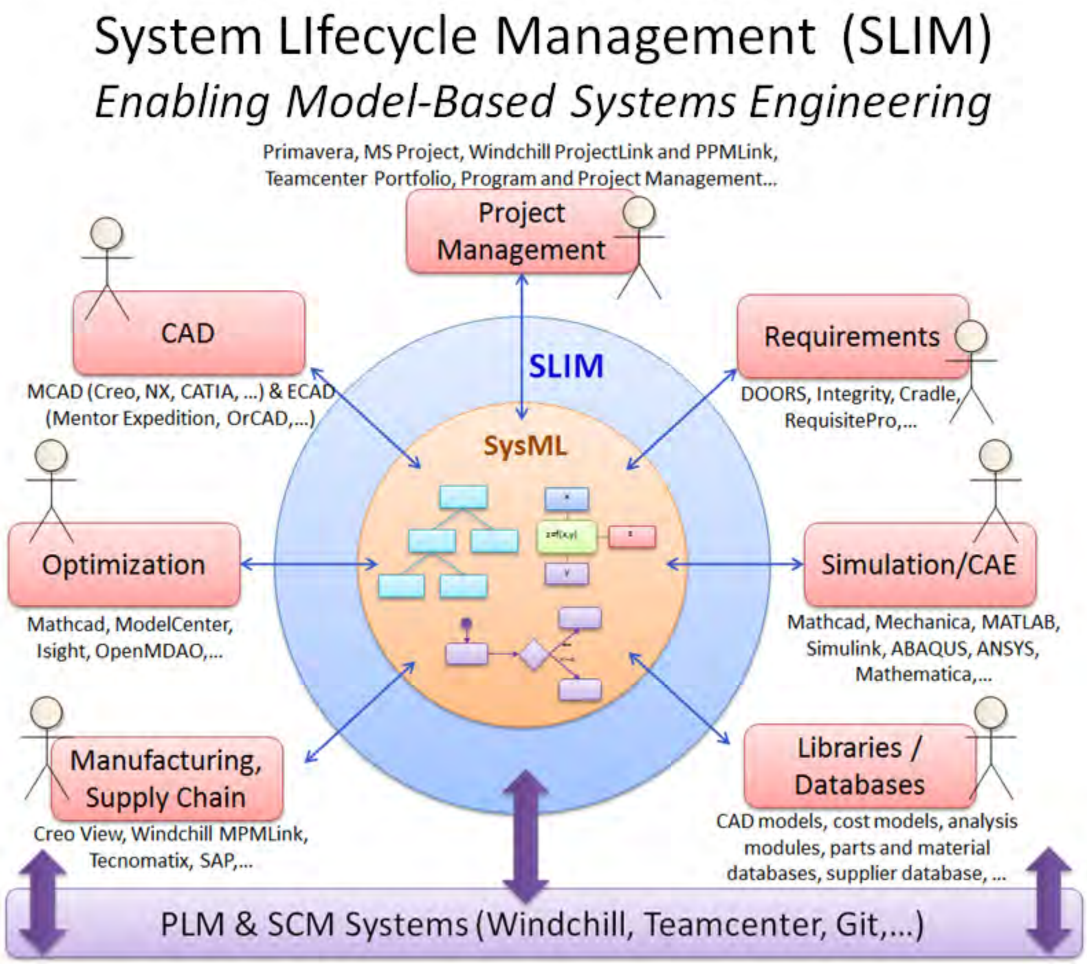

Figure 6. SLIM Concept Diagram (from Intercax 2015)

As another example, Lockheed Martin is attempting to address these challenges by extending the capabilities of MBSE "to support integration across discipline lines" (Oster 2013, 8) including management and customer decision support. Lockheed Martin is employing custom in-house scripts to execute this effort, facilitated by built-in capabilities of existing MBSE products. The objective is to create what Lockheed Martin calls the "model-based program execution" environment (Oster 2013, 12). Integration with PLM, as well as Product Data Management (PDM), is again highlighted as a capability multiplier. Beyond the immediate project, Lockheed Martin suggests that these models can be used to facilitate planning, development, and management of future systems.

INCOSE has created an MBSE Roadmap that shows the path towards full acceptance of the MBSE approach (Figure 6). This roadmap acknowledges the previously identified challenges and the need for maturation of MBSE products. It predicts that MBSE will be fully mature and ready for full adoption at the organizational level by 2020.

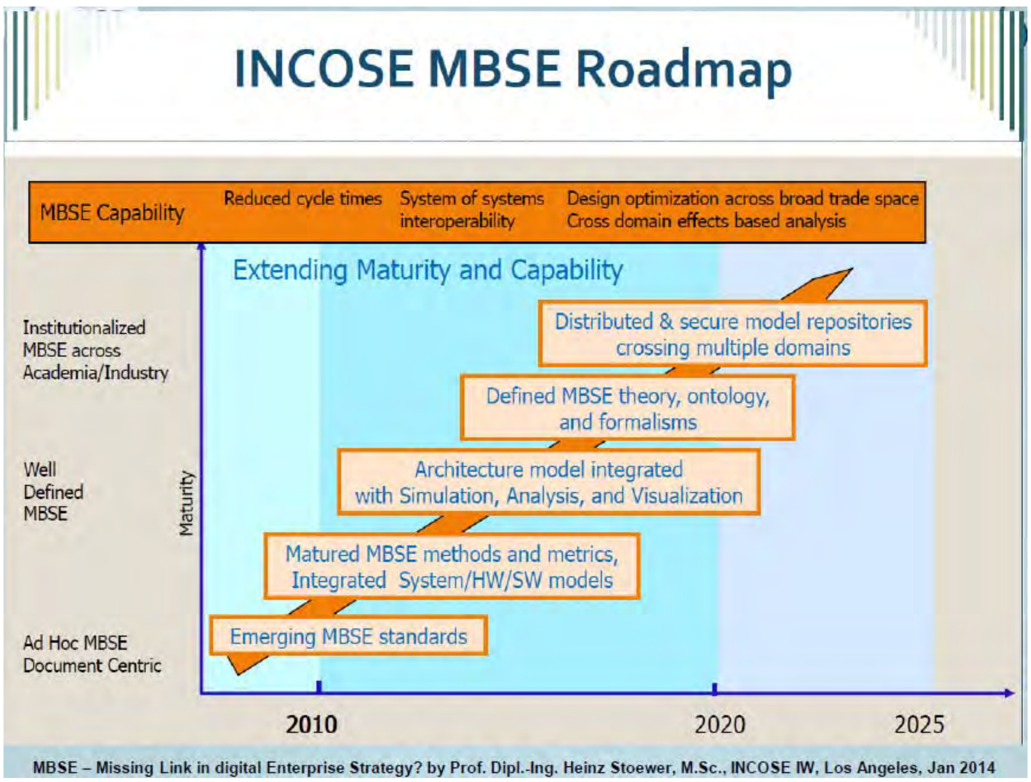

Figure 7. INCOSE MBSE Roadmap (from Heinz 2014, 27)

The DOD has recognized the importance of MBSE, and has created an action in their Acquisition M&S Master Plan (AMSMP) to “Promote model-based systems engineering (MBSE) and M&S-enabled collaborative engineering environments” (DOD 2006, 11). In this same document, the DOD acknowledges the growing importance of MBSE citing the INCOSE Roadmap, growing industry acceptance, and NDIA presentations (Hollenbach 2009, 12). In a separate action, the AMSMP proposes to “support development of open commercial and non-proprietary standards for (model-

based) systems engineering” (Hollenbach 2009, 19), with the goal of assessing for the purpose of implementation within the DOD.

The MBSE community of practice has also recognized the importance of tailoring MBSE products to the DOD. The Object Management Group (OMG) has developed the Unified Modeling Language (UML) 2 standard in order to “enable practitioners to express Department of Defense Architecture Framework (DODAF) and Ministry of Defence Architecture Framework (MODAF) model elements and organize them in a set of specified viewpoints and views that support the specific needs of stakeholders in the U.S. Department of Defense and the United Kingdom Department of Defence” (OMG 2012, 3).

As a specific example of embracing MBSE within the DOD, Defense Information Systems Agency (DISA) has piloted several projects using MBSE. It is currently in the process of transitioning all projects to be supported by MBSE and updating internal systems engineering processes. It is also training its personnel in MBSE. (Okon and Gedo, 9).

B. PRODUCT LIFE CYCLE MANAGEMENT

Produce Life Cycle Management (PLM) is defined as “a systematic, controlled concept for managing and developing products and product related information” (Saaksvuori and Immonen 2008, 3). It is “a holistic concept developed to manage a product and its life cycle including not only items, documents, and Bill of Materials (BOMs), but also analysis results, test specifications, environmental component information, quality standards, engineering requirements, change orders, manufacturing procedures, product performance information, components suppliers, and so forth” (Saaksvuori and Immonen 2008, 2) and includes “workflow, program management, and project control features that standardize, automate, and speed up product management operations” (Saaksvuori and Immonen 2008, 2). It is immediately clear from the definition that PLM can serve as a valuable tool for helping manage systems engineering efforts. Although PLM is not a specific software but instead “a business approach that can align and increase the efficiency and effectiveness of activities” (Schindler 2010, 15),

software is a necessary and major component. Therefore, this analysis will focus on PLM software. Explicit benefits and challenges will be described next. Several PLM products include IBM Collaborative Life Cycle Management, Siemens Teamcenter, and PTC Windchill.

The website PLM Info provides the following list of PLM software benefits:

- Faster time-to-market
- Improved cycle times
- Fewer Errors
- Less scrap & rework
- Greater productivity
- Greater Design efficiency
- Better product quality
- Decreased cost of new product introduction
- Insight into critical processes
- Better reporting and analytics
- Standards and regulatory compliance
- Improved design review and approval processes
- Improved communication
- Reduced product cost and greater profitability
- Better resource utilization
- Improved integration and communication with extended supply chain. (PLM Info 2011).

All of these are desirable from a management standpoint. The three main considerations of management—cost, schedule, and performance—are represented throughout. Communication is highlighted, as well as resource utilization and productivity, all-important components of effectively leading a technical team. Design review and approval is highlighted as well—a key consideration in systems engineering. Also highlighted is better reporting and analytics. The promise is that by ensuring a single common source of data more accurate and timely reports can be generated, and decision makers can be better informed.

In a separate list, John Stark Associates provides the top ten business reasons for implementing PLM (John Stark Associates and SofTech 2007).

1. Get product data under control—Product development is messy; clean it up.
2. Automate product-related processes with workflow for increased productivity—Get rid of the stop-lights.
3. Re-engineer product-related processes—Check for value added and streamline.
4. Reduce product time to market with better application integration—Connect your islands of automation.
5. Develop the right product—Listen to the voice of the customer.
6. Collaboratively develop the best product—Maximize resources, local and global, internal and external.
7. Information reuse—avoid reinventing the wheel.
8. Increase mature product revenues—Listen to the voice of the product.
9. Implement a global product strategy with PLM—Maximize revenues with localized products.
10. Improve product visibility—Manage more effectively with PLM information.

Stark expands on item 3 by stating "PLM brings together previously separate and independent processes in an integrated process architecture" (John Stark Associates and SofTech 2007, 3). This lends well to systems engineering considering its process-heavy nature described in Chapter II. The capability to correlate these processes and track interdependencies is critical to success.

Items 4, 7, and 10 focus on gathering, accessing, connecting, utilizing, and displaying data. Information is often recorded on an independent system, and buried so deep that it is difficult to locate, or may have multiple versions and formats floating around. Saaksvuori and Immonen (2008, 94) cite a Coopers & Lybrand study showing that engineers spend 24% of their time sharing and retrieving information, 21% redoing work, and 14% in meetings largely focused on sharing information. This shows there is a significant opportunity to improve efficiency by integrating applications and supporting reuse—two strengths of PLM systems. Another organizational level advantage stemming

from this improved control of data is “realized when lessons learned from the first generation are applied to all subsequent generations” (Schindler 2010, 17). A system engineering manager would significantly benefit during project startup as well as all future phases from such a data repository of previous work, best practices, and lessons learned.

There are also a number of challenges associated with PLM. The following is a list of challenges presented at a “Beyond PLM” panel discussion at the Aras Community Events International conference in 2011 (Shilovitsky 2011, 6).

- Cost of implementation is too high.
- Cost of change is skyrocketing.
- New platforms need to be validated.
- Customers is [*sic*] demanding vertical solution.
- PLM without PLM is getting some votes.

Additionally, PLM software can significantly “burden [the] organization and people” (Shilovitsky 2011b). There remain a number of challenges related to full integration of PLM software that need to be addressed.

A study by CIMdata, which claims to be the leader in PLM education, research, and strategic management consulting, explored the results that the Aerospace and Defense Industry was seeing from implementing PLM. The research showed that despite heavy PLM investment there were, “with only a few exceptions, uninspiring results” (CIMdata 2013, 1). The study identified two groups: Followers, making up the majority and receiving little value from PLM, and Leaders, making up the minority and receiving significant value. Figure 8 shows how each of these groups viewed the importance of various challenges to the success of implementing PLM in their organizations.

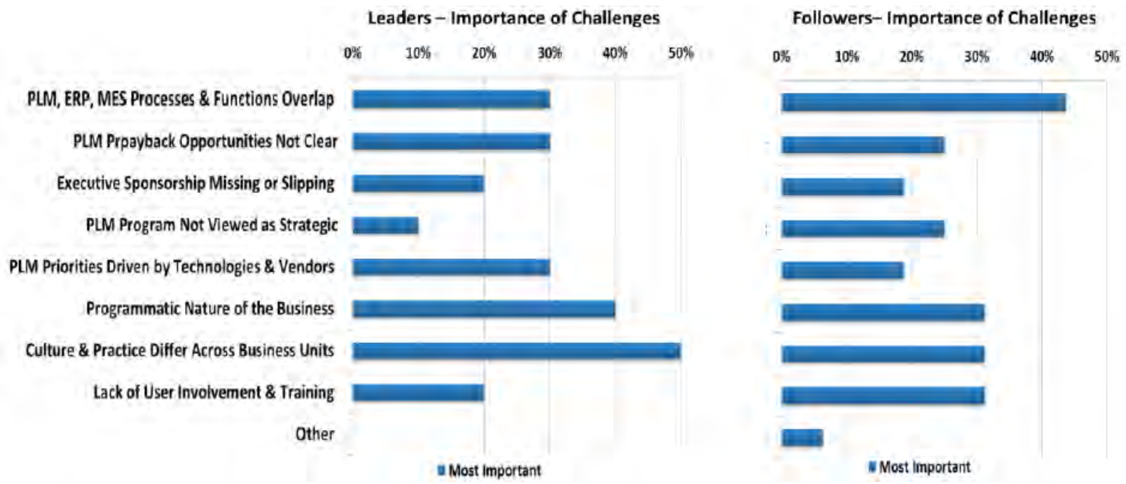

Figure 8. Importance of Challenges to success of implementing PLM in organization, divided among leaders and followers (from CIMdata 2013, 9)

The study highlighted that those organizations seeing little value from their PLM solution found the biggest challenge to be processes and functional overlap with other existing enterprise tools. In contrast, those receiving significant value out of PLM found the biggest challenge to be the culture within, and standardization across, the organization. These, along with the other challenges listed, can all be considered standard challenges when implementing any new system, especially a new systems that is expensive, enterprise-wide, and significantly affects the way business is done.

The DOD is looking to PLM as one of the solutions to deal with “ever-more complex development and support environment…rapidly evolving technologies and threats… [and] higher dependence upon fast-moving commercial technologies”(Borek 2008, 22). The same source concludes that “PLM is a DOD priority” (Borek 2008, 23). There is a specific Integrated Data Environment requirement in the DOD 5000.02 and the Defense Acquisition Guide (DAG) explicitly advocates for an Integrated Data Environment (IDE)/PLM system as part of the systems engineering Technical Data Management Process (DAU 2013).

In response to this push from the DOD the Navy’s Program Executive Office (PEO) Integrated Warfare Systems (IWS) is developing the Enterprise Product Life Cycle Management Integrated Data Environment (ePLM IDE) (Marshall and Murphy

2011). This solution “bridges the gap between the engineering product development and life-cycle product support worlds with a robust ‘enabling’ environment by leveraging a suite of COTS PLM technologies” (Marshall and Murphy 2011, 6). Figure 9 shows the conceptual architecture. It shows ePLM IDE filling a central role in systems engineering management, collaboration, and decision support as it interfaces with systems engineering tools as well as other common tools and products. To further support this initiative, “NAVSEA and DISA have established a Partnership Portfolio allowing for COSTCO pricing” (Smith 2011, 4). This should help overcome two significant challenges: high cost of PLM products, and multiple instantiations of IDE/PLM solutions where a single enterprise solution would be more economical and provide greater capabilities (Smith 2011).

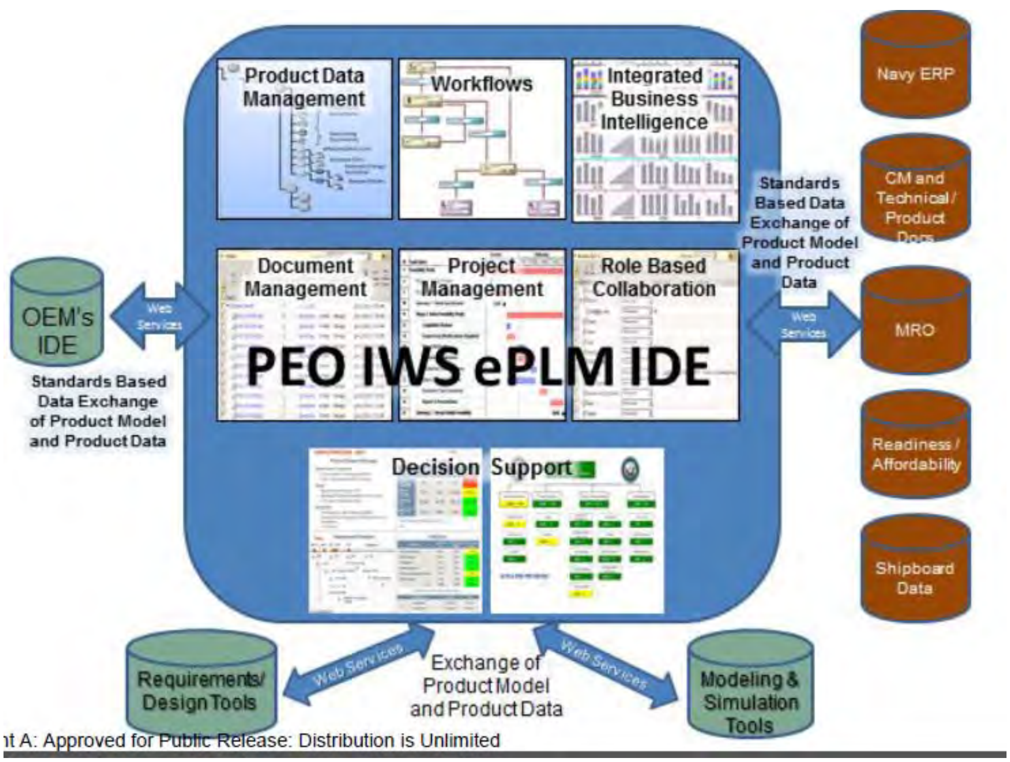

Figure 9. Program Executive Office Integrated Warfare Systems (PEO IWS) ePLM IDE Vision Architecture (from Marshall and Murphy 2011, 5)

C. SYSTEMS ENGINEERING ENVIRONMENT

The complexity of systems engineering is driving the industry to create an integrated environment for executing a systems engineering effort throughout the life cycle. There does not seem to be an industry standard term for these integrated environments, but one common term often used by INCOSE and product developers such as Eclipse and Holagent is Systems Engineering Environment (SEE). Eclipse has developed the Open Systems Engineering Environment (OSEE) and has provided the following definition, which does a good job summarizing the purpose of a SEE.

> The Open System Engineering Environment (OSEE) project provides a tightly integrated environment supporting lean principles across a product's full life-cycle in the context of an overall systems engineering approach. The system captures project data into a common user-defined data model providing bidirectional traceability, project health reporting, status, and metrics which seamlessly combine to form a coherent, accurate view of a project in real-time. By building on top of this data model, OSEE has been architected to provide an all-in-one solution to configuration management, requirements management, testing, validation, and project management. All of these work together to help an organization achieve lean objectives by reducing management activities, eliminating data duplication, reducing cycle-time through streamlined processes, and improving overall product quality through work flow standardization and early defect detection. (Eclipse 2013)

INCOSE has also focused on building a CONOPS and set of requirements (both currently unpublished and in draft) for what it terms the Integrated Systems Engineering Environment (ISEE). The following definition is from a draft ISEE overview document being developed by the INCOSE Tools Interoperability and Integration Working Group ISEE (also unpublished and in draft), and reproduced here by permission of the author.

> the purpose of the Integrated Systems Engineering Environment (ISEE) is to create the computer-aided setting which enables the engineering teams to perform the major functions of Systems Engineering encompassing the entire program life cycle including the management, organization, and technical aspects of systems engineering…The ISEE will eventually address interfaces to other tool environments supporting other facets of program development. (Nallon 2004, 1)

Figure 9, reproduced here by permission of the author, provides an overview of what would be part of ISEE, as well as external interfaces.

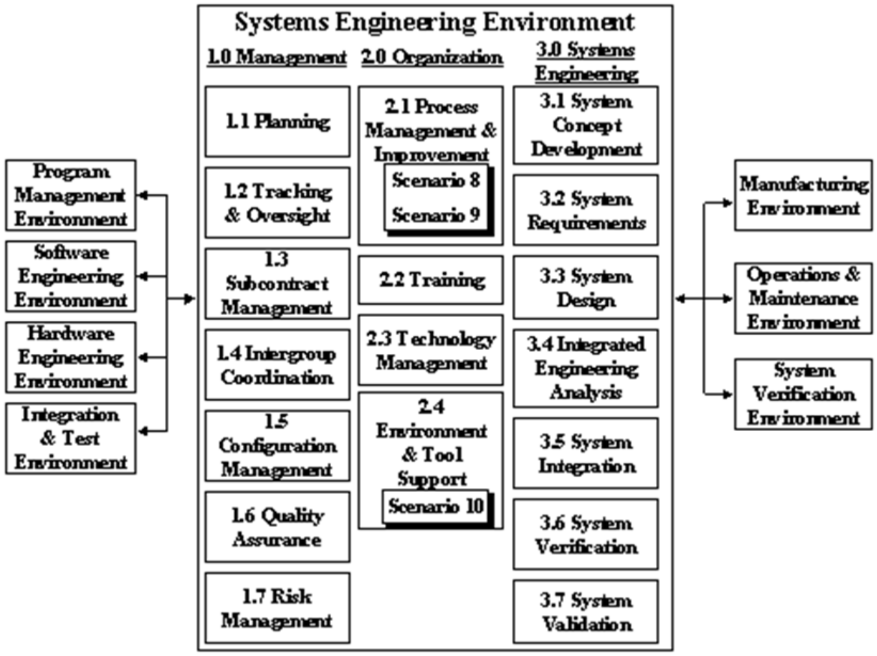

Figure 10. ISEE Functions and Interfaces (from Nallon 2004, 2)

The key message in both of these definitions is that the goal of SEE is to capture all systems engineering efforts and interfaces in a comprehensive and cohesive fashion. This would allow the CSE to manage ongoing work while planning for the entire product life-cycle. Several SEE products include OSEE, 3SL Cradle, and Holagent RDD-100.

Eclipse, the OSEE developer, offers up the following benefits of an SEE (Eclipse 2014):

- support for all engineering aspects (requirements, code, test, project management)
- tightly integrated toolset
- collaborative solution
- consistent user interface across engineering areas
- phased approach for development and extension
- processes integrated into toolset
- decreased cost of all stages of the development life cycle.

All of these support systems engineering management. In fact, management of the systems engineering effort is explicitly included as part of the SEE. The integration of processes into the toolset is also a major benefit from the management perspective. Since systems engineering management is focused on executing and overseeing specific processes, having those already built into the tool increases the probability of success. Finally, SEE improves collaboration across all aspects of systems engineering that can significantly reduce miscommunication and rework, both major obstacles to success as seen in the previous section.

Two additional benefits are worth noting. The first is that the SEE lends itself well to creating integrated dashboard views. These views are geared to quickly extract relevant information and can be customized as needed. This is especially relevant for systems engineering management since the CSE needs to keep track of the big picture on a regular basis and in real-time. Since the SEE tracks all aspects of the ongoing systems engineering effort, as well as the interfaces, it should have sufficient data to build appropriate dashboard views. As an example, 3SL Cradle allows for customized dashboard views by defining key performance indicators and setting thresholds (Figure 11). According to 3SL “This allows managers to manage by exception, so that they can quickly assess the state of the project” (3SL 2015).

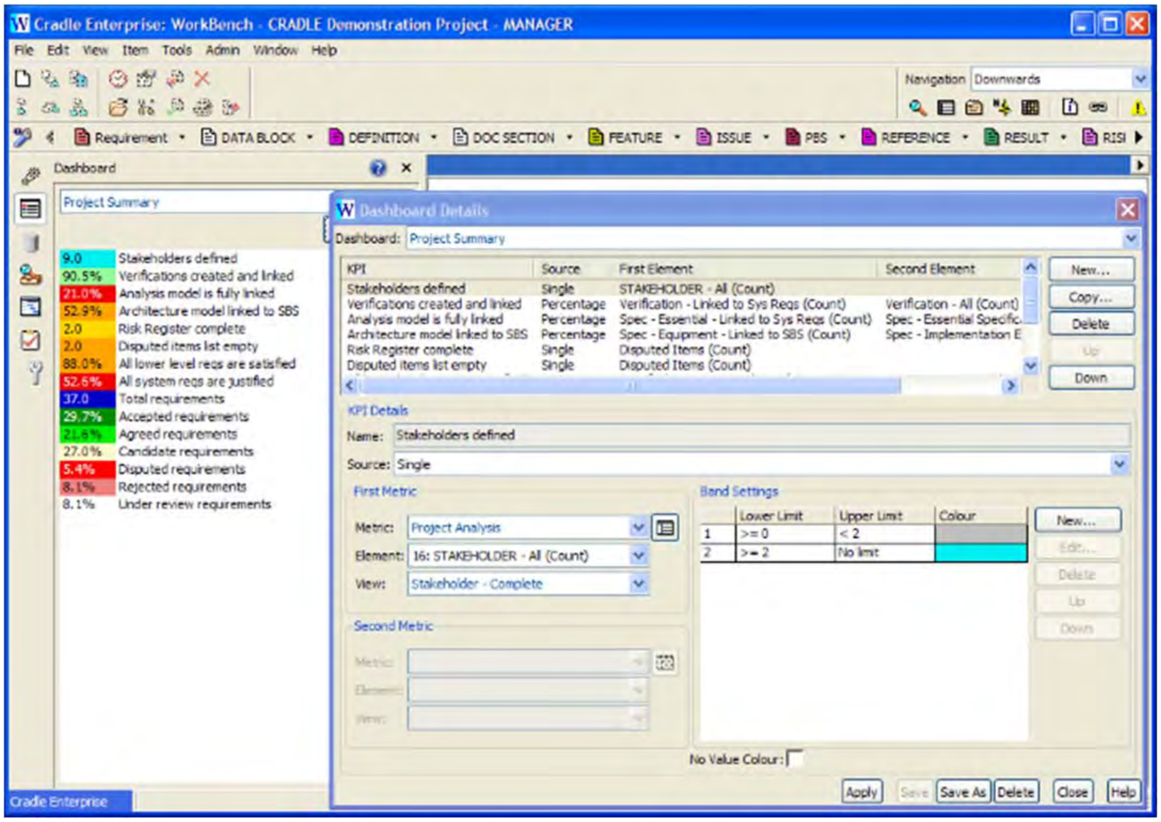

Figure 11. 3SL Cradle Dashboard Customization (from 3SL 2015)

The second additional benefit is that the SEE can be developed to allow for integration with existing tools. This allows the systems engineering team to utilize the preferred tool for a specific function and ensure that the data is also captured within the SEE to maintain big picture awareness. 3SL Cradle shows this integration of tools in Figure 12. One thing to notice is that Cradle interfaces with MBSE and PLM products so that all three of these powerful tools can be used in unison.

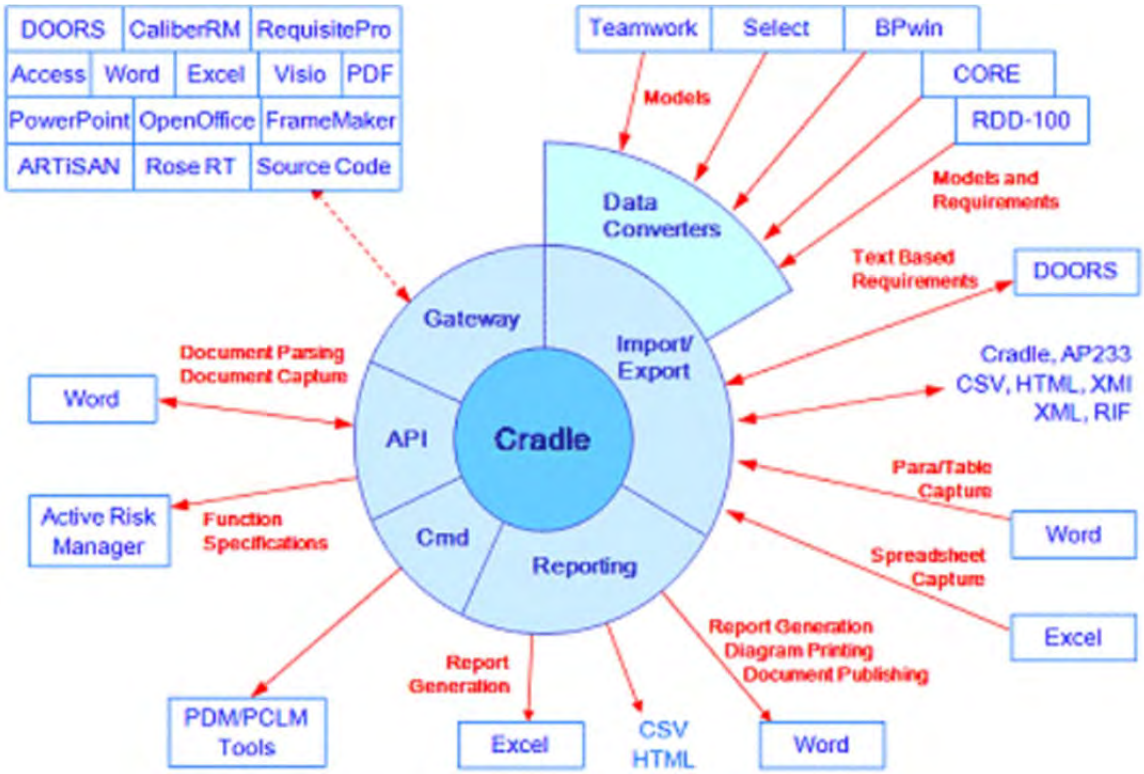

Figure 12. 3SL Cradle Tool Integration (from 3SL 2015)

There are a number of challenges associated with the SEE. One challenge is that due to the large array of projects it is not possible to build a one-size-fits-all product. Therefore, although SEE is supposed to be a "one stop shop" it is unlikely that an SEE product out of the box will contain all the necessary capabilities to make this possible. Therefore, additional work will be required to fill in the gaps. Fortunately, some SEE developers are taking this into account by providing the capability to extend the existing toolset for a particular application. For example, "OSEE contains an Eclipse extension point that allows features to be added to OSEE without having to rebuild the application" (Eclipse 2010). Therefore, the capability to customize the SEE for a specific project does exist.

A second challenge is related to tool integration. As mentioned earlier, SEE depends on the ability to integrate with existing tools. If a specific tool is required for a project and the SEE product does not interface with it that would necessitate either spending significant money to integrate the tool or to leave that tool as stand-alone

product thereby losing some of the advantage of the SEE. In order to help address this challenge, the ISO 10303-AP233 was developed to standardize "representation of systems engineering data" (ISO 2012). That is a big step toward helping to build integrated tools but is merely the first step and requires tool manufacturers to adopt and utilize the standard in their development.

Unfortunately, it appears that SEE has not been able to gain a meaningful foothold within DOD. The only publically available evidence of SEE implementation within the DOD that the author has located is the use of RDD-100 within the Navy Theater Wide Theater Ballistic Missile Defense (NTW TBMD) Program (Hyer and Jones 2000). For this program the ISEE database was segmented into five process areas (requirements, functional behavior, physical architecture, verification methodology, and cost) which were linked together to allow full traceability (Hyer and Jones 2000). And eventually "a strong cornerstone was established by the efforts to establish the requirements in the database and produce a series of reports, traceability matrices, and…a copy of the Systems Requirements Document" (Hyer and Jones 2000). However, no further evidence could be found of the ultimate success of this or any similar DOD efforts which leads the author to believe that establishment of a SEE capability within the DOD has not yet been successful.

D. PROJECT MANAGEMENT TOOLS

The first three categories included either systems engineering specific tools or those that are very closely tied to systems engineering. This last category will focus on a range of tools that, although they do not directly relate to systems engineering, have a number of features that would prove useful to any team and manager. They come from two categories: project management software and social workflow software. Although these are distinct categories there is so much feature overlap that for the purposes of this study we will treat them together. In this category, this focus will be on the benefits and not on the challenges.

Several products in this category include Kenesto, Sparqlight, Asana, AtTask, Base Camp, Red Mine, Deltek's Axium, and Logic Software's Easy Projects. Some of the key features offered by Kenesto (Kenesto 2014) are:

- project workspaces
- dashboards and reports
- document management and vaulting
- cloud document editing
- flexible workflow management
- task management and execution
- drawing and document view and mark-up
- enterprise-class file synchronization
- forms and data management
- data hierarchies.

Task management, dashboards, and workspaces will be addressed in more detail. A common approach for task management seems to be to assign ad-hoc tasking at regular meetings or over email and then wait and hope that this tasking is both understood and fully completed by the required due date. This can often lead to misunderstandings and delays. With the size and complexity of most systems engineering efforts, tasking needs to be formalized to a great extent to be consistently successful. A tasking software solution goes a long way towards accomplishing these objectives and should be a pre-requisite for managing any systems engineering project.

Customizable and personalized dashboards are another key feature that would prove very valuable. CSEs seem to spend much of their time gathering and combining data in order to understand the current status of various efforts and then spend additional time forming that status into reports for their management and stakeholders. As with tasking, the data gathering stage usually consists of individual and team meetings and e-mails which have the drawbacks of being time-consuming, non-real-time, and poorly documented. A dashboard on the other hand provides a more formal and real-time mechanism to gather status on key focus areas and metrics and create reports quickly.

Dashboards also allow easy communication with the project manager and higher-level management.

Finally, workspaces are a key collaboration tool that would provide extreme benefit to the CSE, the systems engineering team, and other stakeholders. The key objective of workspaces is to facilitate communication and teamwork among team-members, managers, and stakeholders in a way that makes it both fast and easy while creating a formal record that can be referenced in the future. It provides a medium to link multiple conversations, actions, and tasks that would normally take place through email, ad-hoc discussions, and team meetings and may not be easily connected otherwise.

E. SUMMARY

This chapter provides a survey of four different categories of software tools that could support systems engineering management. Each category is described and the benefits and challenges are discussed. The first category is MBSE. It is a highly process focused technique that parallels the systems engineering processes. INCOSE predicts that MBSE will be fully mature and ready for full adoption at the organizational level by 2020 and there are DOD efforts underway to embrace MBSE. The second category is PLM. It is a holistic approach for managing systems engineering efforts through the entire life cycle. The DOD is looking at PLM as a solution to help deal with significant complexity and to reduce costs.

The third category is SEE. An SEE is an integrated environment for executing systems engineering efforts throughout the life cycle. The use of a SEE seems very promising but unfortunately does not seem to have been able to gain a meaningful foothold within DOD. The final category is Project Management tools. It contains a range of tools that, although do not directly relate to systems engineering, have a number of features that would prove useful to any team and manager.

All four categories of tools offer features of significant benefit to a CSE. Some of these tools can also be used in combination to extend those benefits (such as MBSE and PLM). And the SEE concept presents a promising approach to having a central system through which the CSE can manage the systems engineering effort. However, there

currently does not seem to be a consolidated commercially available tool or system that allows for seamless management of systems engineering projects across all of the process areas.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. TOOL FEATURES

As discussed in Chapter I, a systems engineering management tool is critical for successful systems engineering management. Although there are multiple tools available, as shown in Chapter III, no current commercially available product addresses all of the systems engineering processes in a consolidated and complete manner. In SDEA, Montgomery, Carlson, and Quartuccio (2012, 13) note that "The challenge is to provide the DOD engineering community an "engineering system" based upon many of these existing tools, coupled with tailored tools which will provide a more integrated repeatable, quantifiable process rather than continuing with the disjointed tool sets and ad-hoc processes." The "engineering system" does not need to be a single product (although it can be), but if not, it does need to be able to combine the use of multiple tools into a single system.

One approach to accomplish this, as discussed in Chapter III when reviewing SEE, is to build a central tool that guides the CSE through the systems engineering processes and is capable of exchanging information with existing tools. This approach is in line with what NDIA notes as one of the top systems engineering issues in a 2010 report, which is the need to "Develop a recommended template for presenting key systems engineering information, including activities, value/expected results, risk of not performing the activities, and future consequences" (NDIA 2010, 7). The tool would act as the master platform for developing, gathering, and presenting key systems engineering information. This chapter describes the high-level requirements for such a tool.

The requirements development approach proposed is to start with the systems engineering processes. Since these engineering processes form the pillars of systems engineering they make a logic starting point for any tool that is intended to guide the systems engineering effort. Furthermore, since the system engineering processes apply to any systems engineering effort they would allow the maximum flexibility to support a broad range of projects. Tailoring would allow the tool to better fit the uniqueness of each project. Such a tool, with the capability to tailor to each project, could prove especially

valuable to DOD acquisition projects that vary significantly but all require a very rigorous adherence to processes per the DOD Acquisition Framework (DODAF).

A. APPROACH

As discussed in Chapter II, systems engineering consistency and completeness rely heavily on standardization provided by processes. Therefore, it seems the natural starting point for a set of tool requirements should be these processes. In Chapter II, two sets of systems engineering processes were explored, DAU and INCOSE. Since the DAU processes are undergoing a major revision at the time of this writing, the below requirements set uses INCOSE processes as the starting point. The sets of processes are close enough, as indicated by the processes mappings presented in Chapter II, that differences in the resulting requirements should not be overly significant. The additional benefit of using INCOSE processes is that the INCOSE Systems Engineering Handbook clearly decomposes each process into activities and sub-activities. This makes it easier to trace to more detailed requirements.

Several stipulations are in order. First, the management tool is intended to be a guide for the CSE and not a replacement for activities and decisions that must still be made by humans. Therefore, not every aspect of every activity or sub-activity can be supported by a requirement. In some instances the tool will only be able to provide a minor contribution in supporting a particular activity or sub-activity. Next, the set of requirements here is not an exhaustive set but is intended as a starting point. Finally, it is important to acknowledge that the challenge of tool integration is a significant one and will not be addressed here beyond stating the need for such integration. As discussed in Chapter III the AP-233 standard does help address this challenge

B. REQUIREMENTS

Below is the high-level decomposition for the management tool (Figure 13). It will help provide the structure for the requirements set. The processes are directly extracted from the INCOSE System Engineering Handbook (INCOSE 2011) and rearranged.

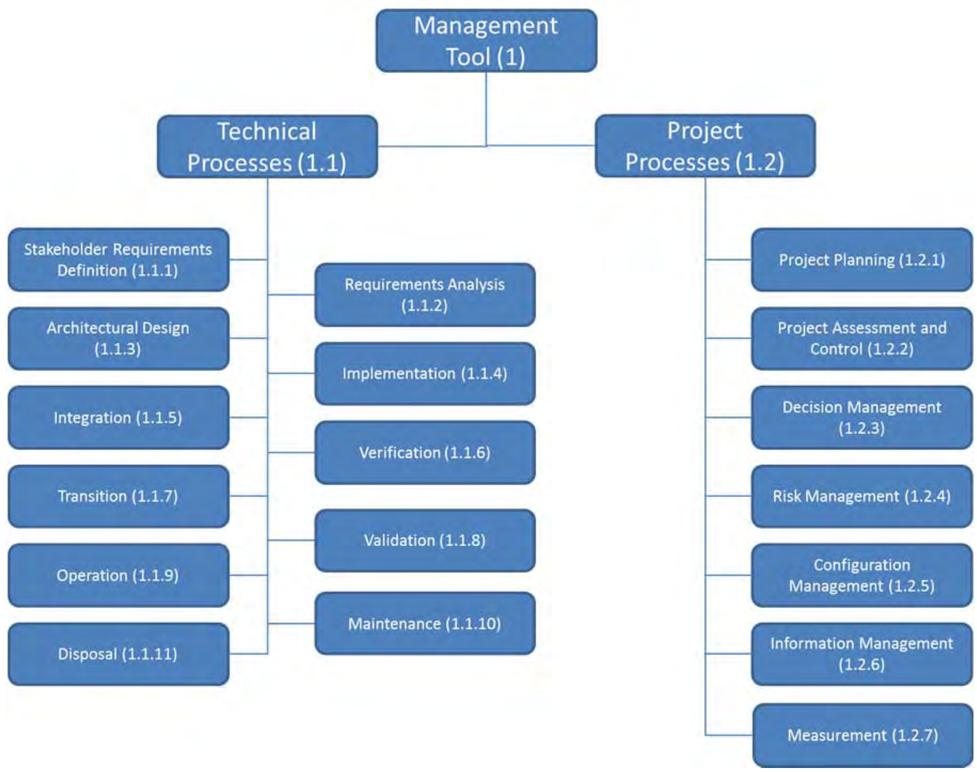

Figure 13. Decomposition

The requirements are listed in the Appendix, starting with top-level requirements, then technical process requirements, and finally project process requirements. For technical process and project process requirements the process activity and next level of detail (here termed sub-activity) from which each requirement is derived are shown. These activities and sub-activities are extracted from the INCOSE Systems Engineering Handbook (INCOSE 2011, Ch 4-5) and reproduced here by permission from INCOSE. The process activities and sub-activities are being treated as the user needs and requirements are traced from these needs. The requirements are developed based on the author's experience as well as insight gained through performing research for Chapters II and III. Some requirements are inspired by the capabilities of existing tools outlined in Chapter II as well as tools the author is familiar with. The remainder of this section will highlight the key features of the set of requirements provided in the Appendix:

- templates
- full traceability

- auto-generated aids
- documentation of results/data
- data review/analysis
- link key internal/external documents
- historical database access
- maintain history
- build and execute scenarios/simulations
- auditing
- access controls.

The following discussion takes a deeper look into each key feature.

1. Templates

Templates are one of the most significant features of the envisioned management tool. Templates would guide the systems engineering team in performing common analysis or developing documents. The templates would be based on best practices and lessons learned and would allow for tailoring. One example of a requirement in this category is Requirement 32: The tool shall provide customizable stakeholder identification template. The template could include predefined attributes such as stakeholder, stakeholder category, their priority, their need, the source of their need, and their desirable and undesirable outcomes. Another example is Requirement 125: The tool shall have a template for building a verification plan. Here the outline of the document would be provided as a starting point, with required section titles, a description of the information expected, and all header and footer data. Such templates allow the team to work from proven and endorsed starting point thereby increasing the chance of success. They can also allow the organization to regularly push updates to all users instead of working from a user pull model that may grow out of sync with multiple versions.

2. Full Traceability

Full traceability is another key feature that is critical for successful systems engineering. The goal is to ensure that there is clear traceability from stakeholders' needs to requirements to the design and to verification and validation. Having multiple

disjointed tools that independently track these elements or failing to formally capture this traceability altogether can result in gaps that lead to an end product that does not meet the stakeholders' needs. An example of a requirement in this category is Requirement 76: The tool shall provide traceability between requirements, functions, and system elements. This will help ensure that the design reflects the requirements and the design description is formally captured for future review.

3. Auto-generated Aids

Auto-generated aids are a broad category that would include checklists, forms, task lists, punch lists, reports, and schedule snapshots among others. Pre-loaded templates would be populated with existing information in the tool to support various systems engineering tasks. An example of a requirement in this category is Requirement 276: The tool shall be capable of auto-generating the entrance and exit criteria checklist. The relevant criteria can be quickly extracted from the source document, placed into a checklist format, and provided to the decision maker for the particular event.

4. Documentation of Results/Data

The tool would be capable of recording all relevant information collected during testing, operation, maintenance, and disposal. Documenting this information is critical in identifying trends and supporting good decision making. An example of a requirement in the category is Requirement 200: The tool shall support logging of preventative maintenance actions taken. Having a single consolidated location to log this information would ensure that future preventative maintenance stays on schedule and there is sufficient history on each item.

5. Data Review/Analysis

Data review and analysis serves to aid in processing of data entered into the tool. Data review would be most useful in the development stage by cross-checking design data against guides, best practices, and lessons learned. An example of a requirement in this category is Requirement 40: The tool shall have an automated review feature that identifies poor and inconsistent requirements based on keywords and historical data. The

tool would scan all requirements and flag requirements that utilize certain keywords or have a particular structure known to be an indicator for bad requirements, similar to a grammar check in word processing software. Another example of a requirement in this category is Requirement 179: The tool shall support comparison of operational performance data against design data and highlight areas of concern. The tool would allow for input of operational data and then would regularly compare that data against the design and provide notifications or trends as well as highlight areas where thresholds have been triggered.

6. Link Key Internal/External Documents

Systems engineering efforts usually draw on multiple documents outside of the immediate project. These can include standards, regulations, and guides that are both internal and external to the organization. Linking to these guides within the tool helps minimize the effort of constantly searching for the correct document each time it is needed. An example of a requirement in the category is Requirement 46: The tool shall have the capability to link to government and industry standards databases. Therefore, if a requirement references a government standard a hyperlink can be included to take the user to that specific reference, or to a locally stored copy of the document with the specific sections of relevance highlighted and with project specific comment saved.

7. Historical Database Access

A key way to increase efficiency is to reuse similar products that have proven to be successful. A historical database would allow for a project to obtain insight into similar efforts within the organization to understand how various processes were executed and how products were developed and to re-use elements as applicable. An example of a requirement in the category is Requirement 41: The tool shall have access to a database of historical requirements for similar systems. This would provide a starting point for requirements as well as history on which were successful and which had issues.

8. Maintain History

Maintaining history is a key feature that would allow for retrieval of any portion of a phase, process, or product within the project. It would also include elements such as configuration control of products under baseline and change history. An example of a requirement in the category is Requirement 92: The tool shall support storage of architectural design decision artifacts. All contributing artifacts such as email exchanges, meeting minutes, trade studies, and analysis of alternatives would be linked to the specific configuration item and requirement so that the history of how a design decision was made and supporting description could be retraced. This would minimize the risk of rehashing design decisions after the fact as a result of faulty recollection or change-over of personnel.

9. Build and Execute Scenarios/Simulations

Building scenarios and simulations allows systems engineers to better understand the results of design decisions and obtain higher certainty that the final design will meet stakeholder needs. Having this capability imbedded within the tool would inform key decisions and provide supporting evidence for future reviews and audits. An example of a requirement in the category is Requirement 87: The tool shall provide the capability to compare multiple models against pre-defined selection criteria. Multiple scenarios can be built and compared against each other and the selection criteria. An objective decision can then be made and supporting artifacts are available to show how that decision was reached.

10. Auditing

A key component of ensuring that that products are correct and processes are being adhered to is regular auditing. The tool will be able to trigger random and pre-set audits which can include both automatic and manual checks. An example of a requirement in the category is Requirement 307: The tool shall be capable of auto-generating an audit checklist to evaluate the Risk Management Process. A checklist would be generated based on the guidelines set by the organizational risk management process, and can be tailored to the project. Some of the answers can be auto-generated

based on existing artifacts and others may require manual review. The results would identify areas of potential improvement and metrics can be saved and kept for the life of the project.

11. Access Control

Having access control is critical to ensuring data integrity. Once baselines are established there needs to be assurance that data will not be manipulated without following an established process. An example of a requirement in the category is Requirement 315: The tool shall be capable of implementing access controls for all CM documentation. All documentation that has formally entered CM control must be restricted so that only authorized personal can make modifications.

C. BENEFITS

The envisioned systems engineering management tool would leverage all of the benefits of the tools described in Chapter III by being able either to integrate with those tools or to reproduce the functionality supported by those tools. There are three areas where the described systems engineering management tool would be especially beneficial.

The first benefit comes from providing a standardized approach to managing a systems engineering effort by guiding it from start to finish. This will help normalize for experience level of the CSE and will be especially helpful in developing less experiencing CSEs. In SDEA Montgomery argues that having an integrated engineering system is especially pertinent now since “the workforce experience level will be contracting over the next decade as the baby boomers retire and the younger engineers grow into that role” (Montgomery, Carlson, and Quartuccio 2012, 25). This will also help mitigate the problem of being highly dependent on one or a few members of the team (CSE being the most critical) that has the entire vision in their head by forcing that vision to be captured in the tool.

The second benefit is the improved insight for the CSE, management, and decision makers. This is enabled by being able to capture real-time status of the project at

any time which provides a good summary of progress and challenges. Having good information quickly supports better decisions and allows for identifying and mitigating problem areas.

The third benefit is the ability to support organizational knowledge transfer. The entire project can be captured from beginning to end and then be “re-played” for post-analysis and for teaching purposes. This also supports easier capturing of lessons learned and best practices. There is less dependence on proactive team members sharing information with the organization and more accurate records of successes and failures along the way.

D. SUMMARY

This chapter builds a set of requirements for a central tool that supports systems engineering management. The approach used is to start with the INCOSE systems engineering processes as the central guide for building such a tool. This approach supports a broad range of systems engineering efforts by allowing for significant tailoring. The requirements are derived from the activities and sub-activities described for each processes. Several key stipulations are offered. First, the management tool is intended to be a guide for the CSE and not a replacement for activities and decisions that must still be made by humans. Second, the set of requirements is not an exhaustive set but is intended as a starting point. Final, the challenge of tool integration is recognized but not addressed by these requirements.

The envisioned systems engineering management tool would leverage the benefits of the existing tools described in Chapter III by either integrating with them or offering similar functionality. There are three areas where the tool would be especially beneficial. The first is to provide a standardized approach to managing a systems engineering effort by guiding it from start to finish. This would help normalize for experience level of the CSE and would also reduce dependence on one or a few key individuals. The second benefit is added insight into progress and challenges for the CSE, management, and decision makers by captured real-time status of the project. The third benefit is more complete and reliable organizational knowledge transfer.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION

A. SUMMARY

The objectives of this thesis are to explore the key components of systems engineering management, conduct a survey of existing software tools that can be used to support systems engineering management, and propose requirements for a tool that would facilitate systems engineering management. The following three research questions are addressed.

1. What are the key components of systems engineering management?

In order to address this question the definition of systems engineering is examined. It is shown that systems engineering is an interdisciplinary, holistic approach that requires a systematic and process-heavy implementation for success. Then a survey of the systems engineering processes is performed, relying on INCOSE and DAU processes. By looking at the processes we understand the breadths of responsibility of the CSE and how important it is for the CSE to have a strong grasp of each process at all times. Finally, various software tools that a CSE commonly utilizes as part of the CSE toolbox are examined. It is noted that these tools provide a powerful mix of functionality but lack integration.

2. What software tools are available that could support systems engineering management?

In order to address this question a survey of the different types of software tools that could support systems engineering management is conducted. Four categories of tools are determined to be most relevant and explored in detail. These include MBSE, PLM, SEE, and Project Management. For each category the benefits and challenges are listed from the perspective of supporting systems engineering management. It is determined that although each category provides powerful functionality that can go a long way towards supporting systems engineering management, there is no current commercially available product that addresses all of the systems engineering processes in a consolidated and complete manner.

3. What requirements would an ideal systems engineering management tool have?

In order to help fill the gap identified through the second research question a set of requirements for an ideal systems engineering management tool are proposed. The starting point is the INCOSE processes and requirements are derived from the activities and sub-activities traced to each process. This approach leverages the benefits of existing tools while also contributing additional benefits.

B. RECOMMENDATIONS

Systems engineering is clearly a complex discipline. There is no single consolidated tool or a suite of integrated tools to support the entire systems engineering management effort. Developing such a tool will significantly benefit the systems engineering community. This will also significantly benefit the DOD in executing highly complex systems engineering efforts. However, it seems that the DOD has not yet started adopting SEE types of tool sets. It will be advantageous for the DOD to put a focus on moving in this direction. This could motivate industry to spend more resources on producing a product that could act as the glue for guiding a systems engineering effort. The starting point for such a product is recommended to be the INCOSE or DAU processes, as described in Chapter IV.

C. FUTURE WORK

The requirements developed in this thesis are just a start. There is significant room to further expand and improve upon these requirements. It will also be beneficial to survey practicing CSEs to obtain feedback on useful requirements. The next step would be to create a prototype systems engineering management tool that can be tested on a real project.

APPENDIX. REQUIREMENTS

A. TOP-LEVEL REQUIREMENTS

The requirements listed in Table 5 represent the top-level requirements for the tool. They apply to both project and technical processes. The column labeled "Level" is based on the decomposition in figure 13, and in this case shows that these requirements are all at the top level. The column labeled "R#" indicates the requirement number for each corresponding requirement. The requirements are shown in the last column and are developed by the author.

Table 5. Top-Level Requirements

Level 1	R#	Requirement
1	1	The tool shall provide modules focused on each of the technical processes identified in the INCOSE Systems Engineering Handbook
	2	The tool shall provide modules focused on each of the project processes identified in the INCOSE Systems Engineering Handbook
	3	The tool shall allow for tailoring of processes and the capability to add comments to describe the tailoring
	4	The tool shall have selectable pre-defined life-cycle models that when selected create interdependencies between the processes
	5	The tool shall generate a checklist showing the processes, activities, and sub-activities that require further attention during any particular phase based on the selected life-cycle model
	6	The tool shall provide process definition hyperlinks to DAU, INCOSE, and other reputable systems engineering websites
	7	The tool shall provide the capability to link to external documents hosted online
	8	The tool shall auto-generate review charts based on customizable parameters
	9	The tool shall auto-generate customizable dashboard views to provide status snapshots
	10	The tool shall provide the capability to build and manage Plan of Action and Milestones (POA&Ms) or action item lists for any particular tasking
	11	The tool shall allow for tracking of detailed entrance and exit criteria for any milestone, tollgate, or task
	12	The tool shall provide customizable templates that can be based on DIDs or other standard formats
	13	The tool shall be capable of requesting random audits for project processes per user customizable parameters

Level 1	R#	Requirement
	14	The tool shall be capable of integrating with common e-mail products
	15	The tool shall be capable of integrating with common spreadsheet products
	16	The tool shall be capable of integrating with common presentation products
	17	The tool shall be capable of integrating with common document development products
	18	The tool shall be capable of integrating with common diagram and flowchart development products
	19	The tool shall be capable of integrating with common CAD products
	20	The tool shall be capable of integrating with common Scheduling products
	21	The tool shall be capable of integrating with common Schedule Assessment products
	22	The tool shall be capable of integrating with common EVM products
	23	The tool shall be capable of integrating with common Simulation products
	24	The tool shall be capable of integrating with common Requirements Management products
	25	The tool shall be capable of integrating with common Information Management products
	26	The tool shall be capable of integrating with common Risk Management products
	27	The tool shall be capable of integrating with common MBSE products
	28	The tool shall be capable of integrating with common PLM products
	29	The tool shall be capable of integrating with common Social Workflow products
	30	The tool shall be capable of integrating with common ERP products
	31	The tool shall be capable of integrating with common Project Management products

B. TECHNICAL PROCESS REQUIREMENTS

The requirements in Table 6 are derived from the INCOSE technical processes. Shaded columns include text from (INCOSE 2011) reproduced here by permission of the copyright holder. Columns that are not shaded are the author's work. Each column labeled "Level" is based on the decomposition in figure 13 and identifies the level for each Process, Activity, and Requirement, as appropriate. The column labeled "R#" indicates the requirement number for each corresponding requirement. The requirements are shown in the last column and are derived by the author from each INCOSE Process, Activity, and Sub-activity.

Table 6. Technical Process Requirements (after INCOSE 2011); shaded columns include text reproduced here by permission of the copyright holder.

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
1.1.1	Stakeholder Requirements Definition (INCOSE 2011, 56)	1.1.1.1	"Elicit Stakeholder Requirements" (INCOSE 2011, 59)	1.1.1.1.1	"Identify stakeholders…" (INCOSE 2011, 59)	32	The tool shall provide customizable stakeholder identification template
				1.1.1.1.2	"Elicit requirements…" (INCOSE 2011, 59)	33	The tool shall support virtual working groups
						34	The tool shall allow for creation of external stakeholder accounts
		1.1.1.2	"Define Stakeholder Requirements" (INCOSE 2011, 59)	1.1.1.2.1	"Define constraints imposed by agreements or interfaces with legacy…systems" (INCOSE 2011, 59)	35	The tool shall have customizable templates to list constraints imposed by agreements or legacy interfaces
				1.1.1.2.2	"Build scenarios…" (INCOSE 2011, 59)	36	The tool shall provide scenario builder capability

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						37	The tool shall support building of DODAF and MoDAF views
				1.1.1.2.3	"Establish critical and desired system performance…" (INCOSE 2011, 60)	38	The tool shall allow for identifying thresholds and objectives and linking those to requirements
				1.1.1.2.4	"Establish MOEs and suitability…" (INCOSE 2011, 60)	39	The tool shall allow for identifying Measures of Effectiveness and Measures of Suitability and linking those to requirements
		1.1.1.3	"Analyze and Maintain Stakeholder Requirements" (INCOSE 2011, 60)	1.1.1.3.1	"Analyze requirements for clarity, completeness, and consistency" (INCOSE 2011, 60)	40	The tool shall have an automated review feature that identifies poor and inconsistent requirements based on keywords and historical data
						41	The tool shall have access to a database of historical requirements for similar systems
				1.1.1.3.2	"Negotiate modifications…" (INCOSE 2011, 60)	42	The tool shall support recording of notes and attachment of files to a requirement or set of requirements
						43	The tool shall have a change log to maintain the history of changes for each requirement
				1.1.1.3.3	"Validate, record, and maintain stakeholder requirements throughout the system	44	The tool shall be able to record and maintain multiple levels or requirements

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					life cycle and beyond…” (INCOSE 2011, 60)		
				1.1.1.3.4	“Establish and maintain a traceability matrix…” (INCOSE 2011, 60)	45	The tool shall allow for traceability amongst requirements
1.1.2	Requirements Analysis (INCOSE 2011, 71)	1.1.2.1	“Define the System Requirements” (INCOSE 2011, 71)	1.1.2.1.1	“Selected standards – Identify standards required to meet quality or design considerations…” (INCOSE 2011, 75)	46	The tool shall have the capability to link to government and industry standards databases
						47	The tool shall provide the capability to import/download relevant standards
						48	The tool shall allow for comments and notes on common file formats
						49	The tool shall allow for creation of hyperlinks between requirements and referenced standards
				1.1.2.1.2	“System boundaries – Clearly identify system elements under design control of the project team and/or organization and expected interactions	50	The tool shall have a customizable system boundaries template

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					with systems external to that control boundary…" (INCOSE 2011, 75)		
						51	The tool shall support traceability between system boundaries and Interface Control Documents (ICD)
				1.1.2.1.3	"External interfaces – Functional and design interfaces…" (INCOSE 2011, 75)	52	The tool shall have a customizable external interfaces template
						53	The tool shall support traceability between external interfaces and ICDs
				1.1.2.1.4	"System Functions – Define system functions that the system is to perform" (INCOSE 2011, 75)	54	The tool shall provide the capability to develop a functional decomposition
						55	The tool shall provide the capability to develop Functional Flow Block Diagram (FFBD), N2 diagrams, and similar
				1.1.2.1.5	"Identify all environmental factors…" (INCOSE 2011, 75)	56	The tool shall provide a database of common environmental factors that may affect performance, impact human comfort or safety, or cause human error for similar systems
						57	The tool shall have a customizable environmental factors template

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.2.1.6	"Life-cycle process requirements…" (INCOSE 2011, 76)	58	The tool shall have customizable maintenance and disposal requirements templates
				1.1.2.1.7	"Design considerations…" (INCOSE 2011, 76)	59	The tool shall provide databases of common Human Systems Integration (HSI), security, and environmental impact design considerations for similar systems
						60	The tool shall have customizable HSI, security, and environmental impact templates
				1.1.2.1.8	"Design constraints…" (INCOSE 2011, 76)	61	The tool shall provide databases of common design constraints including physical limitations, manpower, personnel, and other resource constraints on system operations for similar systems
						62	The tool shall have customizable templates for physical limitations, manpower, personnel, and other resource constraints on system operations
						63	The tool shall have customizable templates for external interface constraints
		1.1.2.2	"Analyze and Maintain the System Requirements"	1.1.2.2.1	"Design verification criteria…" (INCOSE 2011, 76)	64	The tool shall allow for definition of requirement verification approach and criteria in parallel with requirement development

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
			(INCOSE 2011, 76)				
				1.1.2.2.2	"Maintain continuity of configuration control and traceability" (INCOSE 2011, 76)	65	The tool shall maintain a history of all requirement changes, including changes to any requirement attributes
						66	The tool shall allow for binning of requirements into customizable bins
						67	The tool shall maintain requirements traceability
						68	The tool shall allow for baselining of requirements beyond which changes require specific user permissions
						69	The tool shall support user accounts with customizable permissions, including a permission that toggles the ability to make requirements changes after a baseline
1.1.3	Architectural Design (INCOSE 2011, 96)	1.1.3.1	"Define the Architecture" (INCOSE 2011, 98)	1.1.3.1.1	"Define a consistent logical architecture…" (INCOSE 2011, 98)	70	The tool shall support building models of the logical architecture
						71	The tool shall provide an environment for building functional decompositions
						72	The tool shall support definition of attributes and interactions amongst

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							functions
				1.1.3.1.2	"Partition system requirements and allocate them to system elements with associated performance requirements…" (INCOSE 2011, 98)	73	The tool shall support building models of the physical architecture
						74	The tool shall provide an environment for building physical decompositions
						75	The tool shall support definition of attributes and interactions amongst system elements
						76	The tool shall provide traceability between requirements, functions, and system elements
						77	The tool shall support linking and analyzing of existing solutions associated with each system element
				1.1.3.1.3	"Identify interfaces and interactions between system elements…and with external and enabling systems" (INCOSE 2011, 98)	78	The tool shall support documenting and building models of interfaces and interactions amongst system elements

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						79	The tool shall support documenting and building models of interfaces and interactions between system elements and external and enabling systems
				1.1.3.1.4	"Define V&V criteria…" (INCOSE 2011, 99)	80	The tool shall support traceability from requirements verification approach and criteria down to the system elements
						81	The tool shall support building models of system element verification
						82	The tool shall provide a database of common verification criteria for similar systems
		1.1.3.2	"Analyze and Evaluate the Architecture" (INCOSE 2011, 99)	1.1.3.2.1	"Evaluate COTS elements for compatibility with the design" (INCOSE 2011, 99)	83	The tool shall support storing and linking of manufacturer spec sheets
						84	The tool shall provide the capability to display requirements by function and system element
						85	The tool shall provide the capability to customize physical elements within the model based on COTS specs and run a model to determine compatibility and performance

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.3.2.2	"Evaluate alternative design solutions…" (INCOSE 2011, 99)	86	The tool shall provide the capability to develop selection criteria and trace it from the source requirements
						87	The tool shall provide the capability to compare multiple models against pre-defined selection criteria
						88	The tool shall provide a template for creating trade studies
				1.1.3.2.3	"Support definition of the system integration strategy and plan…" (INCOSE 2011, 99)	89	The tool shall provide a template for building an integration strategy
						90	The tool shall provide the capability to display all system internal and external interfaces
		1.1.3.3	"Document and Maintain the Architecture" (INCOSE 2011, 99)	1.1.3.3.1	"Document and maintain the architectural design and relevant decisions made to reach agreement on the baseline design" (INCOSE 2011, 99)	91	The tool shall maintain documentation, models, and any additional artifacts that represent the baseline design
						92	The tool shall support storage of architectural design decision artifacts

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.3.3.2	“Establish and maintain the traceability between requirements and system elements” (INCOSE 2011, 99)	93	The tool shall maintain a history of all design decisions
						94	The tool shall maintain a history of all architectural design changes
						95	The tool shall maintain traceability between the requirements and architectural design
						96	The tool shall allow for baselining of the architecture beyond which changes require specific user permissions
						97	The tool shall support user accounts with customizable permissions, including a permission that toggles the ability to make architectural changes after a baseline
1.1.4	Implementation (INCOSE 2011, 115)	1.1.4.1	“Plan the Implementation” (INCOSE 2011, 118)	1.1.4.1.1	“Develop an implementation strategy – define…procedures, tools and equipment…, implementation tolerances, and the means and criteria for auditing	98	The tool shall provide a template for building an implementation strategy

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					configuration…” (INCOSE 2011, 118)		
						99	The tool shall provide the capability to trigger and record random audits of the configuration against the design documentation
		1.1.4.2	“Perform Implementation” (INCOSE 2011, 118)	1.1.4.2.1	“Develop data for training users…for operating and maintaining…” (INCOSE 2011, 118)	100	The tool shall support documenting training and safety information for each system element
						101	The tool shall provide traceability between system elements and related training and safety documentation
				1.1.4.2.2	“Complete detailed product, process, material specifications…and corresponding analysis and produce documented evidence of implementation compliance [including] conduct[ing] peer reviews and	102	The tool shall provide a template for building specifications documents

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					testing…[and] conducting hardware conformation audits…” (INCOSE 2011, 118)		
						103	The tool shall provide traceability between each model and the specifications documents
						104	The tool shall auto-generate implementation compliance checklists from requirements, models, and specifications
				1.1.4.2.3	“Prepare initial training capability and draft training documentation…” (INCOSE 2011, 118)	105	The tool shall provide a template for preparing training documentation, with segmentation between operations and maintenance
						106	The tool shall provide traceability between system elements and training documentation
				1.1.4.2.4	“Prepare hazardous materials log, if applicable” (INCOSE 2011, 118)	107	The tool shall provide a template for preparing hazardous materials logs

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						108	The tool shall provide the capability to link between system elements and their hazardous materials log entries.
				1.1.4.2.5	"Train initial operators and maintainers…" (INCOSE 2011, 119)	109	The tool shall provide a template for a training strategy
						110	The tool shall auto-generate trainer and maintainer checklists from training documentation
						111	The tool shall maintain a list of trained operators and maintainers
1.1.5	Integration (INCOSE 2011, 120)	1.1.5.1	"Plan Integration" (INCOSE 2011, 122)	1.1.5.1.1	"Define the integration strategy" (INCOSE 2011, 122)	112	The tool shall provide a template for building an integration plan
						113	The tool shall allow for segmentation of integration into phases that can have different objectives and can be linked as needed
				1.1.5.1.2	"Schedule integration testing tools and facilities" (INCOSE 2011, 122)	114	The tool shall provide the capability to record and track key testing tools and facilities details, including scheduling
		1.1.5.2	"Perform Integration" (INCOSE 2011, 122)	1.1.5.2.1	"Assemble system elements according to the integration plan" (INCOSE 2011, 122)	115	The tool shall allow regular progress updates by the integration team to track detailed integration status
						116	The tool shall auto-generate proposed tasking lists based on the

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							integration plan and progress updates
						117	The tool shall generate a visualization of the integration progress by annotating system logical and physical architecture models
				1.1.5.2.2	“Validate and verify interfaces…” (INCOSE 2011, 122)	118	The tool shall auto-generate an interfaces checklist with relevant characteristics from the requirements and design documents
						119	The tool shall provide an anomaly tracker
						120	The tool shall have the capability to elevate anomalies and deficiencies and track them through a POA&M
				1.1.5.2.3	“Verify and analyze assemblies…” (INCOSE 2011, 122)	121	The tool shall auto-generate a checklist of functions from the requirements and design documents
				1.1.5.2.4	“Document integration testing and analysis results” (INCOSE 2011, 122)	122	The tool shall provide templates for documenting integration testing and analysis results
				1.1.5.2.5	“Document and control the architectural baseline…” (INCOSE 2011, 122)	123	The tool shall provide the capability to capture and store architectural baselines

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						124	The tool shall provide the capability to execute a formal change process for any architectural baseline modifications
1.1.6	Verification (INCOSE 2011, 126)	1.1.6.1	“Plan Verification” (INCOSE 2011, 128)	1.1.6.1.1	“Schedule, confirm, and install verification enabling systems” (INCOSE 2011, 128)	125	The tool shall provide a template for building a verification plan
						126	The tool shall provide the capability to record and track details related to verification enabling systems, including scheduling and VV&A
						127	The tool shall provide the capability to annotate the logical and physical architecture models to show the verification architecture, including explicitly identifying verification enabling systems
						128	The tool shall provide the capability to document differences between the test environment and operational environment, including capturing a risk assessment of the difference
						129	The tool shall provide the capability to develop high-level verification concepts linked to requirements
		1.1.6.2	“Perform Verification”	1.1.6.2.1	“Develop verification procedures” (INCOSE	130	The tool shall provide a template for building verification procedures

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
			(INCOSE 2011, 128)		2011, 128)		
						131	The tool shall allow for development of a verification test step library, including linking to external test step databases
						132	The tool shall auto-generate verification witness sign-off forms based on configurable parameters
				1.1.6.2.2	"Conduct verification activities…to demonstrate compliance with requirements" (INCOSE 2011, 128)	133	The tool shall be capable of generating day by day schedule snapshots of the verification schedule
						134	The tool shall be capable of capturing daily progress updates and calculating whether the verification activity is on track
				1.1.6.2.3	"Document verification results and enter data into the RVTM" (INCOSE 2011, 128)	135	The tool shall provide the capability to document verification results, including saving red-lined test procedures
						136	The tool shall provide a template for building a verification report
						137	The tool shall link verification results with the requirements database
						138	The tool shall auto-generate an

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							RVTM
1.1.7	Transition (INCOSE 2011, 131)	1.1.7.1	“Plan the Transition” (INCOSE 2011, 134)	1.1.7.1.1	“Prepare a transition strategy, including operator training, logistics support, delivery strategy, and problem rectification/resolution strategy” (INCOSE 2011, 134)	139	The tool shall provide a template for building a training plan
						140	The tool shall provide a template for building an ILS plan (i.e. Life Cycle Support Plan, Integrated Support Plan, etc.)
						141	The tool shall support delivery planning including documenting shipping lead times, action item tracking, and need dates
				1.1.7.1.2	“Develop installations procedures” (INCOSE 2011, 134)	142	The tool shall provide a template for building an installation plan
						143	The tool shall provide a template for building the installation procedures
						144	The tool shall allow linking of installation drawings to installation procedures
		1.1.7.2	“Perform the Transition” (INCOSE 2011,	1.1.7.2.1	“Prepare the installation site and install system…”	145	The tool shall allow for documentation of regular progress updates by the installation team to

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
			134)		(INCOSE 2011, 134)		track detailed installation status
						146	The tool shall generate a visualization of the installation progress by annotating high-level installation drawings
				1.1.7.2.2	"Train the users…and affirm users have the knowledge and skill levels necessary to perform Operation and Maintenance activities." (INCOSE 2011, 134)	147	The tool shall support development of computer based training modules
						148	The tool shall support linking each system element to any existing training materials (i.e. COTS and Government Off the Shelf (GOTS) training materials)
						149	The tool shall support development of operator and maintainer prerequisites checklists
				1.1.7.2.3	"Receive final confirmation that the system meets…[user's] needs. This process typically ends with a formal, written	150	The tool shall auto-generate a list of all applicable documents (as customizable by the user) that support successful delivery of system

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					acknowledgement…" (INCOSE 2011, 134)		
				1.1.7.2.4	"Post-implementation problems are documented and may lead to corrective actions or changes to the requirements" (INCOSE 2011, 134)	151	The tool shall provide a template for documenting post-implementation problems and linking to affected requirements and action items
1.1.8	Validation (INCOSE 2011, 135)	1.1.8.1	"Plan Validation" (INCOSE 2011, 137)	1.1.8.1.1	Develop a validation strategy (INCOSE 2011, 137)	152	The tool shall provide a template for building a validation plan
						153	The tool shall provide the capability to annotate the logical and physical architecture models to show the validation architecture
						154	The tool shall provide the capability to develop and model assessment scenarios
		1.1.8.2	"Perform Validation" (INCOSE 2011, 137)	1.1.8.2.1	"Develop validation procedures…" (INCOSE 2011, 137)	155	The tool shall provide a template for building validation procedures
						156	The tool shall allow for development of a validation test step library, including linking to external test step

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							databases
						157	The tool shall auto-generate validation witness sign-off forms based on configurable parameters
				1.1.8.2.2	"Ensure readiness to conduct validation…" (INCOSE 2011, 137)	158	The tool shall allow for tracking of entrance criteria
						159	The tool shall provide the capability to record and track details related to validation enabling systems, including scheduling and VV&A
				1.1.8.2.3	"Support in-process validation throughout the system development" (INCOSE 2011, 137)	160	The tool shall link user generated validation considerations to each of the following technical processes: requirements analysis, architectural design, implementation, integration, verification, and transition
				1.1.8.2.4	"Conduct validation to demonstrate conformance to stakeholder requirements" (INCOSE 2011, 138)	161	The tool shall be capable of generating day by day schedule snapshots of the validation schedule
						162	The tool shall be capable of capturing daily progress updates and calculating whether the validation activity is on track

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.8.2.4	“If anomalies are detected, analyze for corrective actions and detect trends in failures…” (INCOSE 2011, 138)	163	The tool shall provide a template for recording anomalies
						164	The tool shall support generation of anomaly burn-down POA&Ms
						165	The tool shall provide templates for troubleshooting techniques, such as fishbone diagrams, that can be linked to anomalies
						166	The tool shall have the capability to plot failures over time and against specific configuration items or subsystems to support failure trend analysis
				1.1.8.2.5	“Recommend corrective actions and obtain stakeholder acceptance of validation results” (INCOSE 2011, 138)	167	The tool shall support documenting of corrective actions for each anomaly
						168	The tool shall support documenting of a regression test plan for each anomaly
				1.1.8.2.6	“Document validation results and enter data into the RVTM”	169	The tool shall provide the capability to document validation results, including saving red-lined test

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 138)		procedures
						170	The tool shall provide a template for building a validation report
						171	The tool shall link validation results with the requirements database
1.1.9	Operation (INCOSE 2011, 139)	1.1.9.1	"Prepare for Operations" (INCOSE 2011, 141)			172	The tool shall provide a template for building a concept of operations
						173	The tool shall support building models to visualize the Concept of Operations and scenarios
						174	The tool shall provide a template for a training package, and link to DOD and service specific training standards
		1.1.9.2	"Perform Operational Activation and Check-out" (INCOSE 2011, 141)	1.1.9.2.1	"Provide operator training and maintain qualified staff" (INCOSE 2011, 141)	175	The tool shall support development of an operator training plan and task list
						176	The tool shall support development of an operator qualifications checklist
		1.1.9.3	"Use System for Operations"	1.1.9.3.1	"Execute ConOps for the system-of-	177	The tool shall auto-generate execution templates from the

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
			(INCOSE 2011, 141)		interest" (INCOSE 2011, 141)		Concept of Operations and operator task list
				1.1.9.3.2	"Track system performance and account for operational availability" (INCOSE 2011, 141)	178	The tool shall perform operational availability calculations based on issue and anomaly data
				1.1.9.3.3	"Perform operational analysis" (INCOSE 2011, 141)	179	The tool shall support comparison of operational performance data against design data and highlight areas of concern
						180	The tool shall support comparison of operational cost data against design data and highlight areas of concern
		1.1.9.4	"Perform Operational Problem Resolution" (INCOSE 2011, 141)	1.1.9.4.1	"Manage operational support logistics" (INCOSE 2011, 141)	181	The tool shall support documenting operational issues and anomalies
				1.1.9.4.2	"Document system status and actions taken" (INCOSE 2011, 141)	182	The tool shall support regular logging of system status and actions taken
				1.1.9.4.3	"Report malfunctions and make recommendations for improvement"	183	The tool shall support recording of malfunctions and auto-generate recommendations based on a look-up database

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 141)		
						184	The tool shall support linking to a database of malfunctions and corrective actions
		1.1.9.5	“Support the Customer” (INCOSE 2011, 141)			185	The tool shall be capable of generating tailored operation status reports for a specific period of time or for the life of the system
						186	The tool shall be capable of pushing regular operation status updates to the customer
1.1.10	Maintenance (INCOSE 2011, 142)	1.1.10.1	“Plan Maintenance” (INCOSE 2011, 144)	1.1.10.1.1	“Establish a maintenance strategy” (INCOSE 2011, 144)	187	The tool shall have a template for building a maintenance strategy
						188	The tool shall support development of a maintainer training plan and task list
						189	The tool shall support documenting of maintenance actions for each configuration item
				1.1.10.1.2	“Define maintenance constraints on the system requirements” (INCOSE 2011, 144)	190	The tool shall allow for linking of maintenance constraints to system requirements

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.10.1.3	"Obtain the enabling systems, system elements, and other services used for maintenance of the system" (INCOSE 2011, 144)	191	The tool shall support documentation and tracking of all maintenance agreements and highlight upcoming renewal dates
						192	The tool shall support logging and tracking of maintenance enabling systems
				1.1.10.1.4	"Monitor replenishment levels of spare parts" (INCOSE 2011, 144)	193	The tool shall track all spare parts and locations
						194	The tool shall perform spare levels calculations to support the required operational availability
						195	The tool shall maintain a list of vendors and estimated lead time for all spares
				1.1.10.1.5	"Manage the skills and availability of trained maintenance personnel" (INCOSE 2011, 145)	196	The tool shall support development of a maintainer qualifications checklist
						197	The tool shall maintain a list of all qualified maintenance personnel

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
		1.1.10.2	Perform Maintenance	1.1.10.2.1	"Implement maintenance and problem resolution procedures…" (INCOSE 2011, 145)	198	The tool shall support development of a preventative maintenance schedule and highlight near term and late tasks
						199	The tool shall auto-generate a list of maintenance activities for each configuration item
				1.1.10.2.2	"Maintain a history of failures, actions taken, and other trends…" (INCOSE 2011, 145)	200	The tool shall support logging of preventative maintenance actions taken
						201	The tool shall be capable of generating a list of preventative maintenance actions taken and plot against time
						202	The tool shall support logging of all failures
						203	The tool shall be capable of generating a list of historical failures and plot against time
				1.1.10.2.3	"Monitor customer satisfaction with system and maintenance support" (INCOSE 2011, 145)	204	The tool shall be capable of generating tailored support status reports for a specific period of time or for the life of the system
						205	The tool shall be capable of pushing regular support status updates to the customer

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
1.1.11	Disposal (INCOSE 2011, 145)	1.1.11.1	"Plan Disposal" (INCOSE 2011, 148)	1.1.11.1.1	"Review the Concept of Disposal…" (INCOSE 2011, 148)	206	The tool shall have a template for documenting all hazardous materials
				1.1.11.1.2	"Define the Disposal Strategy" (INCOSE 2011, 148)	207	The tool shall have a template for building a disposal strategy
						208	The tool shall support documentation of required deactivation, disassembly, and removal steps for each element
				1.1.11.1.3	"Impose associated constraints on the system requirements" (INCOSE 2011, 148)	209	The tool shall allow for linking of disposal constraints to system requirements
		1.1.11.2	"Perform Disposal" (INCOSE 2011, 148)	1.1.11.2.1	"Deactivate the elements to be terminated" (INCOSE 2011, 148)	210	The tool shall auto-generate a checklist for deactivation of each element
						211	The tool shall auto-generate procedures for deactivation of each element
				1.1.11.2.2	"Disassemble the elements for ease of handling" (INCOSE 2011, 148)	212	The tool shall auto-generate a checklist for disassembly of each element
						213	The tool shall auto-generate procedures for disassembly of each element

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.1.11.2.3	“Remove the elements and any associated waste products from the operational site” (INCOSE 2011, 148)	214	The tool shall auto-generate a checklist for removal of each element
						215	The tool shall auto-generate procedures for removal of each element
		1.1.11.3	“Finalize the Disposal” (INCOSE 2011, 148)	1.1.11.3.1	“Maintain documentation of all Disposal activities and residual hazards” (INCOSE 2011, 148)	216	The tool shall support documenting all disposal activities taken
						217	The tool shall support tracking of all hazardous material from removal to disposal

C. PROJECT PROCESS REQUIREMENTS

The requirements in Table 7 are derived from the INCOSE project processes. Shaded columns include text from (INCOSE 2011) reproduced here by permission of the copyright holder. Columns that are not shaded are the author's work. Each column labeled "Level" is based on the decomposition in figure 13 and identifies the level for each Process, Activity, and Requirement, as appropriate. The column labeled "R#" indicates the requirement number for each corresponding requirement. The requirements are shown in the last column and are derived by the author from each INCOSE Process, Activity, and Sub-activity.

Table 7. Project Process Requirements (after INCOSE 2011); shaded columns include text reproduced here by permission of the copyright holder.

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
1.2.1	Project Planning (INCOSE 2011, 178)	1.2.1.1	Define the Project (INCOSE 2011, 182)	1.2.1.1.1	"Analyze the project proposal and related agreements to define the project scope" (INCOSE 2011, 182)	218	The tool shall link to a database of previous proposals
						219	The tool shall provide a template for building a scope document (i.e. Statement of Work (SOW))
				1.2.1.1.2	"Identify project objectives and project constraints" (INCOSE 2011, 182)	220	The tool shall provide a template for documenting project constraints and objectives

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.2.1.1.3	“Establish tailoring of organization procedures and practices to carry out planned effort” (INCOSE 2011, 182)	221	The tool shall link to organizational procedures and practices
						222	The tool shall provide a tailoring wizard to tailor organizational procedures and practices and output the tailored document
				1.2.1.1.4	“Define and maintain a life cycle mode that is tailored from the defined life cycle models of the organization” (INCOSE 2011, 182)	223	The tool shall link to organizationally defined life-cycle models
						224	The tool shall provide a tailoring wizard to tailor organizationally defined life-cycle models and output the tailored model
						225	The tool shall support tracking of progress against the tailored organizational life-cycle model by linking to progress for each process

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
		1.2.1.2	“Plan Project Resources” (INCOSE 2011, 182)	1.2.1.2.1	“Establish the roles and responsibilities for project authority” (INCOSE 2011, 182)	226	The tool shall provide a template for building the project organizational hierarchy
						227	The tool shall provide a template for building a project management plan
						228	The tool shall provide roles and responsibilities templates for each role defined in the project organizational hierarchy, including touch points between positions
						229	The tool shall generate position descriptions from the defined roles and responsibilities to support hiring
				1.2.1.2.2	“Define top-level work packages for each task and activity…[and tie] to required resources and procurement strategies” (INCOSE 2011, 182)	230	The tool shall link to the required work package structure either per the organization or per the contract
						231	The tool shall provide a template for populating each work package with detailed tasks and activities

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						232	The tool shall provide the capability to link work packages to the project schedule
						233	The tool shall provide the capability to identify resources (including manpower and cost) for each work package
				1.2.1.2.3	"Develop a project schedule based on objectives and work estimates" (INCOSE 2011, 182)	234	The tool shall provide the capability to build a resource loaded project schedule
						235	The tool shall provide the capability to link project schedule tasks to project objectives (i.e., explicit SOW tasks)
				1.2.1.2.4	"Define the infrastructure and services required" (INCOSE 2011, 182)	236	The tool shall provide a template for defining the required infrastructure and services (i.e.. facilities, contracts support, IT, etc)
				1.2.1.2.5	"Define costs and estimate project budget" (INCOSE 2011, 182)	237	The tool shall provide a template for building a Basis of Estimate, based on the scope and work packages and linked to the project schedule
				1.2.1.2.6	"Plan the acquisition of materials, goods and enabling systems services"	238	The tool shall support linking of acquisition of materials, goods, and enabling systems to the Basis of Estimate and project schedule

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 182)		
		1.2.1.3	Plan Project Technical and Quality Management	1.2.1.3.1	"Prepare a Systems Engineering Plan (SEP); tailor the Quality, Configuration, Risk and Information Management plans to meet the needs of the project" (INCOSE 2011, 182)	239	The tool shall provide a template for building a SEP
						240	The tool shall provide templates for building the QA, Configuration Management (CM), Risk, and Information Management plans
						241	The tool shall link to a database of QA, CM, Risk, and Information Management plans and provide a template for tailoring of those plans
				1.2.1.3.2	"Tailor the organizational Risk Management Processes and practices in accordance with	242	The tool shall provide a tailoring wizard to tailor organizational risk management processes and output the tailored document

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					the agreements and the SEP…” (INCOSE 2011, 182)		
				1.2.1.3.3	“Tailor the organizational Configuration Management Processes and practices in accordance with the agreements and the SEP…” (INCOSE 2011, 182)	243	The tool shall provide a tailoring wizard to tailor organizational configuration management processes and output the tailored document
		1.2.1.4	“Activate the Project” (INCOSE 2011, 183)			244	The tool shall link to any organizational tools or enterprise systems to allow for formal activation of the project
1.2.2	Project Assessment and Control (INCOSE 2011, 197)	1.2.2.1	“Assess the Project” (INCOSE 2011, 201)	1.2.2.1.1	“Determine actual and projected cost against budget, actual and projected time against schedule, and deviations in project quality” (INCOSE 2011, 201)	245	The tool shall provide a template for developing a project controls strategy

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						246	The tool shall support the capability to implement EVM
						247	The tool shall be capable of comparing actual and projected costs against budget using either EVM or user configurable metrics
						248	The tool shall be capable of comparing actual and projected progress against the project schedule using either EVM or user configurable metrics
						249	The tool shall be capable of documenting user customizable quality metrics and comparing against plans
				1.2.2.1.2	"Evaluate the effectiveness and efficiency of the performance of project activities" (INCOSE 2011, 201)	250	The tool shall generate cost, schedule, and risk progress reports at a user defined frequency
						251	The tool shall support calculation of a Defense Contract Management Agency (DCMA) 14 point schedule assessment and highlight weaknesses
				1.2.2.1.3	"Evaluate the adequacy and the availability of the project infrastructure"	252	The tool shall support documenting of all project infrastructure needs and how they are to be (or are being) met

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 201)		
						253	The tool shall be capable of tracking infrastructure needs and availability, and provide alerts of availability conflicts
				1.2.2.1.4	"Evaluate project progress against established criteria and milestones" (INCOSE 2011, 201)	254	The tool shall be capable of displaying cost and schedule progress to any user defined milestone
						255	The tool shall track satisfactory completion of contractual items and requirements and be able to generate displays showing this progress
				1.2.2.1.5	"Conduct required reviews, audits, and inspections to determine readiness to proceed to next milestone" (INCOSE 2011, 201)	256	The tool shall support user configurable review, audit, and inspection templates
						257	The tool shall link to review, audit, and inspection guidance from the organization and contract

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						258	The tool shall support linking of reviews, audits, and inspections to milestone entrance and exit criteria
				1.2.2.1.6	"Monitor critical tasks and new technologies…" (INCOSE 2011, 201)	259	The tool shall support identification of a critical path
						260	The tool shall support identification of critical tasks for heightened monitoring
				1.2.2.1.7	"Make recommendations for adjustments to project plans…"(INCOSE 2011, 201)	261	The tool shall auto-generate areas of concern based on schedule, cost, and performance progress
				1.2.2.1.8	"Communicate status as designated in agreements, policies, and procedures" (INCOSE 2011, 201)	262	The tool shall provide report templates based on organizational and contractual requirements
						263	The tool shall auto-populate reports based on user configurable parameters and links
				1.2.2.1.9	"Analyze assessment results" (INCOSE	264	The tool shall support user configurable displays of project controls assessment results

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					2011, 201)		
		1.2.2.2	"Control the Project" (INCOSE 2011, 201)	1.2.2.2.1	"Initiate corrective actions when assessments indicate deviation from approved plans" (INCOSE 2011, 201)	265	The tool shall provide a list of corrective action suggestions
						266	The tool shall support user configurable triggers for plan deviations and provide an alert
				1.2.2.2.2	"Initiate preventive actions when assessments indicate a trend toward deviation" (INCOSE 2011, 201)	267	The tool shall provide a list of preventive action suggestions
						268	The tool shall support user configurable triggers for deviation trends and provide an alert
				1.2.2.2.3	"Initiate problem resolution when assessments indicate non-conformance with performance success criteria"	269	The tool shall provide a problem resolution template that includes performance success criteria

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 201)		
				1.2.2.2.4	“Establish work items and changes to schedule to reflect actions taken” (INCOSE 2011, 201)	270	The tool shall support development and tracking of a problem resolution POA&M
				1.2.2.2.5	“Negotiate with suppliers for any goods or services acquired from outside the organization” (INCOSE 2011, 201)	271	The tool shall provide a template for supplier agreements
						272	The tool shall be capable of tracking all suppliers and supplier agreements
				1.2.2.2.6	“Make the decision to proceed, or not to proceed, when assessments support a tollgate or milestone event” (INCOSE 2011, 201)	273	The tool shall link to all entrance and exit criteria for all tollgate and milestone events from the organization and contract

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						274	The tool shall support tailoring of the entrance and exit criteria
						275	The tool shall support linking each entrance and exit criteria to specific documents, models, or any other data contained within the tool or linked to the tool
						276	The tool shall be capable of auto-generating the entrance and exit criteria checklist
						277	The tool shall be capable of providing suggestions of what artifacts are commonly used for a particular entrance or exit criteria
						278	The tool shall be capable of providing an assessment whether all entrance or exit criteria are linked to an artifact
		1.2.2.3	"Close the Project" (INCOSE 2011, 201)			279	The tool shall link to any organizational tools or enterprise systems to allow for formal close out of the project
1.2.3	Decision Management (INCOSE 2011, 202)	1.2.3.1	"Plan and Define Decisions" (INCOSE 2011, 204)	1.2.3.1.1	"Identify the need for a decision and the strategy for making the decision, including desired outcomes and measureable success criteria"	280	The tool shall provide a decision analysis resolution template, which includes measureable success criteria

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 204)		
						281	The tool shall support setting of triggers for commencing the decision analysis resolution process
		1.2.3.2	"Analyze the Decision Information" (INCOSE 2011, 204)	1.2.3.2.1	"Involve all personnel with knowledge and experience relevant to the decision" (INCOSE 2011, 204)	282	The tool shall provide a list of recommended participants based on the decision category and the project organizational hierarchy chart
				1.2.3.2.2	"Evaluate the consequences of alternative choices using the selected strategy and optimize the decision" (INCOSE 2011, 205)	283	The tool shall support analysis of alternative choices through weighted ratings
						284	The tool shall support simulation of decisions with impacts on cost, schedule, performance, and risk
				1.2.3.2.3	"Make the decision, based on the relevant data	285	The tool shall rank the choices based on the results of the weighted ratings

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					and inputs" (INCOSE 2011, 205)		
		1.2.3.3	"Track the Decision" (INCOSE 2011, 205)	1.2.3.3.1	"Record the decision, with the relevant data and supporting documentation" (INCOSE 2011, 205)	286	The tool shall support documenting of the final decision and all relevant data and supporting documentation
				1.2.3.3.2	"Communicate new directions from the decision" (INCOSE 2011, 205)	287	The tool shall update budget, schedule, technical, and risk data based on the decision parameters
1.2.4	Risk Management (INCOSE 2011, 215)	1.2.4.1	"Plan Risk Management" (INCOSE 2011, 218)	1.2.4.1.1	"Define and document the risk strategy" (INCOSE 2011, 218)	288	The tool shall provide a template documenting the risk plan which includes risk, issue, and opportunity management
						289	The tool shall support execution of electronic risk boards
		1.2.4.2	"Manage the Risk Profile" (INCOSE 2011, 218)	1.2.4.2.1	"Define and document risk thresholds and acceptable and unacceptable risk conditions" (INCOSE 2011, 218)	290	The tool shall provide a risk identification form that allows detailed documentation of risks

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						291	The tool shall support user configurable thresholds for likelihood and consequence levels
				1.2.4.2.2	"Periodically communicate the risks (and opportunities) with the appropriate stakeholders" (INCOSE 2011, 218)	292	The tool shall auto-generate risk burn-down charts from the risk identification forms
						293	The tool shall auto-generate various views to visualize the risk profile, including views that show all risks simultaneously from approval to close-out
		1.2.4.3	"Analyze Risks" (INCOSE 2011, 219)	1.2.4.3.1	"Identify and define risk situations" (INCOSE 2011, 219)	294	The tool shall support development of candidate risks
						295	The tool shall support tracking of watch items
				1.2.4.3.2	"Analyze risks for likelihood and consequence to determine the magnitude of the risk and its priority	296	The tool shall auto-generate risk rankings based on the risk exposure (LxC and/or L+C)

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					for treatment" (INCOSE 2011, 219)		
				1.2.4.3.3	"Define a treatment scheme and resources for each risk, including identification of person who will be responsible…" (INCOSE 2011, 219)	297	The tool shall support selection of a treatment scheme (avoid, accept, control, or transfer) for each risk
						298	The tool shall support identification of a POC for each risk, candidate risk, and watch item
		1.2.4.4	"Treat Risks" (INCOSE 2011, 219)	1.2.4.4.1	"Using the criteria for acceptable and unacceptable risk, generate a plan of action when the risk threshold exceeds acceptable levels" (INCOSE 2011, 219)	299	The tool shall support development of a plan of action for each risk that is triggered by the risk thresholds
		1.2.4.5	"Monitor Risks" (INCOSE 2011, 219)	1.2.4.5.1	"Maintain a record of risk items and how they were treated" (INCOSE	300	The tool shall maintain the history of each closed risk

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					2011, 219)		
						301	The tool shall support documentation of all relevant minutes, decisions, and artifacts for each risk, candidate risk, and watch item
						302	The tool shall record progress against all risk milestones and footstones
						303	The tool shall link each risk to the impacted tasks in the project schedule and translate the risk burn-down profile to the most likely, worst case, best case durations for each impacted task
						304	The tool shall support calculation of a schedule risk analysis to any user defined tollgate or milestone
				1.2.4.5.2	"Maintain transparent risk management communications" (INCOSE 2011, 219)	305	The tool shall show all risk working groups and boards on the schedule
		1.2.4.6	"Evaluate the Risk Management Process" (INCOSE 2011, 219)	1.2.4.6.1	"Define, analyze, and document measures indicating the status of the risk and effectiveness of the treatment	306	The tool shall auto-generate summary views of the risk process, including effectiveness of risk treatment

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					alternatives" (INCOSE 2011, 219)		
						307	The tool shall be capable of auto-generating an audit checklist to evaluate the Risk Management Process
1.2.5	Configuration Management (INCOSE 2011, 228)	1.2.5.1	"Plan Configuration Management" (INCOSE 2011, 230)	1.2.5.1.1	"Implement a configuration control cycle that incorporates evaluation, approval, validation, and verification of ECRs" (INCOSE 2011, 230)	308	The tool shall provide a template for a configuration management plan
						309	The tool shall provide a template for a change control process to include management of ECRs
						310	The tool shall support execution of configuration control boards
		1.2.5.2	"Perform Configuration Management" (INCOSE 2011, 231)	1.2.5.2.1	"Configuration Identification - Identify system elements to be maintained under configuration control" (INCOSE	311	The tool shall document system elements to be maintained under configuration control

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					2011, 231)		
				1.2.5.2.2	"Configuration Control - Establish the configuration baselines and control baseline changes throughout the system life cycle" (INCOSE 2011, 231)	312	The tool shall record all baseline data and artifacts associated with each system element under configuration control
						313	The tool shall trigger a configuration control board for any baseline changes and document all relevant data, including the new baseline
				1.2.5.2.3	"Configuration Status Accounting - Develop and maintain configuration control documentation and communicate the status of the controlled items to the project team"	314	The tool shall be capable of storing all configuration control documentation that can be accessed by the team but can only be modified by select personnel

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 231)		
						315	The tool shall be capable of implementing access controls for all CM documentation
				1.2.5.2.4	“Configuration Audits - Perform audits associated with milestones and decision gates to validate the baselines” (INCOSE 2011, 231)	316	The tool shall trigger baseline configuration audits based on decision gates and milestones
						317	The tool shall auto-generate baseline configuration audit checklists
1.2.6	Information Management (INCOSE 2011, 237)	1.2.6.1	“Plan Information Management” (INCOSE 2011, 240)	1.2.6.1.1.	“Supporting establishing and maintaining a system data dictionary…” (INCOSE 2011, 240)	318	The tool shall provide an information repository

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
				1.2.6.1.2	“Define system-relevant information, storage requirements, access privileges, and the duration of maintenance” (INCOSE 2011, 240)	319	The tool shall support establishing access privileges for the information repository
						320	The tool shall provide user configurable attributes for storage requirements that can be applied to each system element and across the system
				1.2.6.1.3	“Define formats and media for capture, retention, transmission, and retrieval of information” (INCOSE 2011, 240)	321	The tool shall support web-based access of data in the information repository
						322	The tool shall support e-mailing of data in the information repository
						323	The tool shall support capture of e-mailed documents into the information repository
				1.2.6.1.4	“Identify valid sources of information”	324	The tool shall provide an attribute to indicate the maturity of any data

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					(INCOSE 2011, 240)		
						325	The tool shall provide an attribute to indicate the source of any data
		1.2.6.2	"Perform Information Management" (INCOSE 2011, 240)	1.2.6.2.1	"Periodically obtain artifacts of information" (INCOSE 2011, 240)	326	The tool shall query for updated documents based on the document delivery dates in the project schedule
						327	The tool shall support sending of internal data update requests
				1.2.6.2.2	"Maintain information according to security and privacy requirements" (INCOSE 2011, 240)	328	The tool shall be capable of maintaining information at multiple levels of sensitivity
						329	The tool shall support security controls for the information repository
				1.2.6.2.3	"Retrieve and distribute information, as required" (INCOSE 2011, 240)	330	The tool shall support queries of the information repository
						331	The tool shall auto-generate information management reports based on user

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							configurable parameters
						332	The tool shall make the information repository accessible to stakeholders, with configurable permissions
				1.2.6.2.4	"Archive designated information for compliance with legal, audit, and knowledge retention requirements" (INCOSE 2011, 240)	333	The tool shall provide user configurable attributes for each artifact that designate archive requirements such as retention duration
				1.2.6.2.5	"Retire unwanted, invalid, or unverifiable information according to organizational policy, security, and privacy requirements" (INCOSE 2011, 240)	334	The tool shall support implementation of an information retirement schedule

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
1.2.7	Measurement (INCOSE 2011, 242)	1.2.7.1	“Plan Measurement” (INCOSE 2011, 245)	1.2.7.1.1	“Establish a measurement strategy” (INCOSE 2011, 245)	335	The tool shall provide a template for a measurement strategy, which will be a subset of the project management plan
				1.2.7.1.2	“Identify the measurement stakeholders” (INCOSE 2011, 245)	336	The tool shall support identification of stakeholders for each measurement
				1.2.7.1.3	“Identify and prioritize the information needs of the decision makers and stakeholders” (INCOSE 2011, 245)	337	The tool shall allow for prioritizing, annotating, and adding identifiers for each data artifact
				1.2.7.1.4	“Identify and select relevant measures that aid with the management and technical performance of the program” (INCOSE 2011, 245)	338	The tool shall allow linking of data artifacts to specific measures
						339	The tool shall suggest common measures used to support management and

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
							technical performance
				1.2.7.1.5	“Define the base measures, derived measures, indicators, data collection, measurement frequency, measurement repository, reporting method and frequency, trigger points or thresholds, and review authority” (INCOSE 2011, 245)	340	The tool shall support defining of user configurable attributes for each measure
		1.2.7.2	Perform Measurement	1.2.7.2.1	“Collect, store and verify the data per plan” (INCOSE 2011, 245)	341	The tool shall support recording of measurement data
				1.2.7.2.2	“Process and analyze the data to obtain measurement results…” (INCOSE 2011, 245)	342	The tool shall support multiple views of measurement data

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
						343	The tool shall support common methods for processing measurement data
				1.2.7.2.3	“Document and review measurement information products with measurement stakeholders and recommend actions” (INCOSE 2011, 245)	344	The tool shall auto-generate charts that show collected measurement data
						345	The tool shall auto-generate various views of the measurement data to identify trends and history
		1.2.7.3	Evaluate Measurement	1.2.7.3.1	“Evaluate the effectiveness of the measures for providing the necessary insight for decisions” (INCOSE 2011, 245)	346	The tool shall track the history of changes to measures
						347	The tool shall make suggestions for changes to measure attributes based on historical results
				1.2.7.3.2	“Evaluate the effectiveness, efficiency, and	348	The tool shall be capable of auto-generating an audit checklist to evaluate the Measurement Process

Level 3	Process (after INCOSE 2011)	Level 4	Activity (after INCOSE 2011)	Level 5	Sub-activity (after INCOSE 2011)	R#	Requirement
					compliance of the Measurement Process” (INCOSE 2011, 246)		
				1.2.7.3.3	‘Assign corrective actions, if required” (INCOSE 2011, 246)	349	The tool shall be capable of linking corrective actions to specific tasks in the project schedule
				1.2.7.3.4	“Document and store all program measures and corrective actions in a measurement repository” (INCOSE 2011, 246)	350	The tool shall store all measurement data, including corrective actions, in the information repository

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

3SL. 2015. “Cradle from Concept to Completion.” Accessed January 22, 2015 http://www.threesl.com/pages/Cradle/English/Content/Products/workbench.php.

Bajaj, Manas, Dirk Zwemer, Russell Peak, Alex Phung, Andrew Scott, and Miyako Wilson. 2011. “4.3.1 Satellites to Supply Chains , Energy to Finance — SLIM for Model-Based Systems Engineering Part 1 : Motivation and Concept of SLIM.” In *INCOSE International Symposium*, 21:368–394. Denver, CO. doi:10.1002/j.2334–5837.2011.tb01214.x. http://www.omgsysml.org/SLIM_for_MBSE_Bajaj_Part1.pdf.

Borek, Kevin. 2008. “The Opportunities and Challenges of Product Life cycle Management Within the Department of Defense.” Presentation at the 3D Collaboration & Interoperability ‘08, Denver, CO. http://catiacommunity.com/pdf/borek.pdf.

CIMdata. 2013. “An Emerging PLM Value Gap in the Aerospace & Defense Industry: Initial Characterization of PLM Value Leaders.” Ann Arbor, MI: CIMdata, Inc. http://www.cimdata.com/en/resources/2-uncategorised/212-an-emerging-plm-value-gap-in-the-aerospace-defense-industry-initial-characterization-of-plm-value-leaders.

Comer, Edward R., and Sharon L. Rohde. 1992. “System Engineering Concept Demonstration, Effort Summary.” RL-TR-92–345. Vol. 1. Griffiss AFB, NY: Rome Laboratory Air Force Materiel Command. www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA265467.

Defense Acquisition University (DAU). 2013. “Chapter 4.3: Systems Engineering Processes.” In *Defense Acquisition Guidebook*. Accessed January 22, 2015 https://acc.dau.mil/CommunityBrowser.aspx?id=638325&lang=en-U.S..

Eclipse. 2010. “OSEE/BLAM.” Accessed January 22, 2015 http://wiki.eclipse.org/OSEE/BLAM.

———. 2013. “OSEE.” Accessed January 22, 2015 http://wiki.eclipse.org/OSEE.

———. 2014. “OSEE/Users Guide/Getting Started.” Accessed January 22, 2015 http://wiki.eclipse.org/OSEE/Users_Guide/Getting_Started#Screencasts.

Friedenthal, Sanford, Regina Greigo, and Mark Sampson. 2007. “INCOSE Model-Based Systems Engineering (MBSE) Initiative.” Presentation at INCOSE 2007, San Diego, CA. https://www.incose.org/enchantment/docs/07Docs/07Jul_4MBSEroadmap.pdf.

Friedenthal, Sanford, Alan Moore, and Rick Steiner. 2012. “Chapter 2: Model-Based Systems Engineering.” In *A Practical Guide to SysML–The Systems Modeling Language*, 15–27. Waltham, MA: Elsevier Inc.

Heinz, S. “Model Based Systems Engineering (MBSE) Missing Link in the Digital Enterprise Strategy?” Presented at the 2014 INCOSE MBSE Workshop, Los Angeles, CA, January 25, 2014. http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:01-iw14-mbse_workshop-keynote-missing_link_in_the_digital_enterprise-a-stoewer.pdf.

Hollenbach, James W. “Execution of the Acquisition M & S Master Plan: Progress Report.” Presented at NDIA Systems Engineering M&S Committee Meeting, Washington D.C., April 22, 2009. http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Committees/M_S Committee/2009/April 2009/10. AMSMP Execution_Hollenbach.pdf.

Hyer, Scott A., and Mark W. Jones. 2000. “1.2.1 Realizing Complete Traceability With An Integrated Systems Engineering Environment (ISEE).” In *INCOSE International Symposium*, 10:143–148. Denver, CO. doi:10.1002/j.2334–5837.2000.tb00369.x.

International Council on Systems Engineering (INCOSE). 2004. “What Is Systems Engineering?” Accessed January 22, 2015 http://www.incose.org/practice/whatissystemseng.aspx.

———. 2011. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Version 3.2.2. San Diego, CA: INCOSE.

ISO/IEC. 2008. *Systems and Software Engineering–System Life Cycle Processes*. Geneva, Switzerland: International Organization for Standardization / International Electrotechnical Commissions. ISO/IEC/IEEE 15288:2008.

Intercax. 2015. “What Is SLIM?” Accessed January 22, 2014 http://www.intercax.com/products/slim.

International Organization for Standardization (ISO). 2012. “ISO 10303–233:2012 Industrial Automation Systems and Integration -- Product Data Representation and Exchange -- Part 233: Application Protocol: Systems Engineering.” ISO. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=55257.

John Stark Associates and SofTech, Inc. 2007. “Top 10 Business Reasons for Implementing PLM.” SofTech, Inc. http://www.softech.com/plm-white-papers/top-10-business-reasons-for-implementing-plm.

Kendrick, Tom. 2009. *Identifying and Managing Project Risk*. 2nd ed. New York, NY: AMACOM.

Kenesto. 2014. “Kenesto’s Capabilities.” Accessed January 22, 2015 http://www.kenesto.com/kenestos-capabilities.

Marshall, Jolene, and Thomas Murphy. “PEO IWS Enterprise Product Life cycle Management Integrated Data Environment (ePLM IDE).” Presented at 2011 Integrated Warfare Systems Conference, Washington D.C.: Program Executive Office (PEO) Integrated Warfare Systems (IWS). http://www.dtic.mil/ndia/2011PEO/JimmySmith.pdf.

Montgomery, Paul, Ron Carlson, and John Quartuccio. 2012. “System Definition-Enabled Acquisition (SDEA)—A Concept for Defining Requirements for Applying Model-Based Systems Engineering (MBSE) to the Acquisition of DOD Complex Systems.” In *Ninth Annual Acquisition Research Symposium*. Vol. 1. Monterey, CA. http://hdl.handle.net/10945/33858.

Nallon, John F. 2004. “ISEE Volume 1.” INCOSE Tools Integration & Interoperability Working Group Working Paper available to INCOSE members (unpublished).

National Defense Industrial Association (NDIA). 2008. “Report on Systemic Root Cause Analysis Of Program Failures.” Washington, DC: U.S. Department of Defense. http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Studies/NDIASRCAReportFINA18Dec2008.pdf.

———. 2010. “Top Systems Engineering Issues In U.S. Defense Industry.” Washington D.C. http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Studies/Top SE Issues 2010 Report v11 FINAL.pdf.

———. 2011. “Final Report of the Model Based Engineering (MBE) Subcommittee.” Washington D.C.: National Defense Industrical Association. http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Committees/M_S Committee/Reports/MBE_Final_Report_Document_(2011–04–22)_Marked_Final_Draft.pdf.

Object Management Group (OMG). 2012. “Unified Profile for DODAF and MODAF (UPDM).” Object Management Group. http://www.omg.org/spec/UPDM/2.0/PDF.

Okon, Walt, and Chris Gedo. “DISA’s Transition to Model Based Systems Engineering (MBSE).” Washington D.C.: Defense Information Systems Agency (DISA). http://www.disa.mil/news/conferences-and-events/customer-and-industry-forum-2011/~/media/files/disa/news/conference/2011/joint-enablers/transition_model_based_systems_engineering.ppt.

Oster, Christopher. “So I’ve Modeled…What’s Next? Extending the Use of Systems Models in an Engineering Enterprise.” Presented at INCOSE IW 2013 MBSE Workshop, Jacksonville, FL, January 26–27, 2013. Lockheed Martin. http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:02–2013_incose_mbse_workshop-i_ve_modeled_so_what_s_next-oster.pdf.

PLM Info. 2011. “Product Life Cycle Management: What Is PLM?” Accessed January 22, 2015 http://www.product-lifecycle-management.info/what-is-plm/plm-benefits.html.

Saaksvuori, Antti, and Anselmi Immonen. 2008. *Product Life cycle Management*. Third. Berlin, Germany: Springer.

Schindler, Christopher M. 2010. “Product Life cycle Management: A Collaborative Tool for Defense Acquisitions.” Naval Postgraduate School.

Shenhar, Aaron J., and Brian Sauser. 2009. “Systems Engineering Management: The Multidisciplinary Discipline.” In *Handbook of Systems Engineering and Management, 2nd Edition*, 2nd ed., 117–154. Hoboken, NJ: John Wiley & Sons. January 18, 2014. http://books.google.com/books?hl=en&lr=&id=eFRwQuzPnEcC&oi=fnd&pg=PA117&dq=Systems+Engineering+Management+:+The+Multidisciplinary+Discipline&ots=ffctOY9SCH&sig=CbwXR7L3L0qnL47tDHnJD0WXThQ.

Shilovitsky, Oleg. 2011a. “Beyond PLM: What Will Impact PLM in the next 10 Years?” Accessed January 22, 2015 http://www.slideshare.net/olegshilovitsky/beyondplm-panel-discussion-on-ace-2011#.

———. 2011b. “What Is the Biggest PLM Challenge?” Accessed January 22, 2015 http://beyondplm.com/2011/09/06/what-is-the-biggest-plm-challenge/.

Smith, Jimmy D. “PEO IWS Enterprise Product Life cycle Management Integrated Data Environment (ePLM IDE).” Presented at Surface Navy Association Open Architecture Forum, San Diego, CA, August 16, 2011. http://media.navysna.org/OA/SMithPEOIWS.pdf.

U.S. Department of Defense (DOD). 2006. “Acquisition Modeling and Simulation Master Plan.” Washington D.C.: U.S. Department of Defense. http://www.acq.osd.mil/se/docs/AMSMP_041706_FINAL2.pdf.

U.S. Government Accountability Office (GAO). *Managing Risk to Achieve Better Outcomes*. (GAO-10–374T). Washington, DC: GPO, 2010. http://www.gao.gov/assets/130/123946.pdf.

Vitech Corp. “Model Based Systems Engineering.” Presented at Asia Pacific Council of Systems Engineering (APCOSE) 2011, Seoul, South Korea, October 19–21. http://www.incose.org/chesapek/Docs/2011/Presentations/2011_09_17_Model-Based SystemsEngineeringPublicSlides.pdf.

Zipes, Lori F. “Program Management vs Systems Engineering: How different are they?” Presented at 10th Annual Systems Engineering Conference “Focusing on Improving Performance of Defense Systems Programs,” San Diego, CA, October 22–25, 2007.
http://www.dtic.mil/ndia/2007systems/Wednesday/AM/Track3/5381.pdf.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California